

**LINEAMIENTOS PARA GENERACIÓN DE CONTENIDOS
EDUCATIVOS Y RUTAS DE APRENDIZAJE PERSONALIZADOS
PARA EL PROYECTO E-LANE EN LA UNIVERSIDAD DEL
CAUCA**



**CARLOS ANDRÉS LUCERO AGUIRRE
DIEGO FERNANDO PINO MOSQUERA**

**Monografía presentada como requisito para optar al título de Ingeniero en
Electrónica y Telecomunicaciones**

Director: Ing. MARIO FERNANDO SOLARTE SARASTY

Universidad del Cauca
Facultad de Ingeniería Electrónica y Telecomunicaciones
Departamento de Telemática
Popayán, Junio de 2006

**LINEAMIENTOS PARA GENERACIÓN DE CONTENIDOS
EDUCATIVOS Y RUTAS DE APRENDIZAJE PERSONALIZADOS
PARA EL PROYECTO E-LANE EN LA UNIVERSIDAD DEL
CAUCA**

**CARLOS ANDRÉS LUCERO AGUIRRE
DIEGO FERNANDO PINO MOSQUERA**

**Monografía presentada como requisito para optar al título de Ingeniero en
Electrónica y Telecomunicaciones**

Director: Ing. MARIO FERNANDO SOLARTE SARASTY

Universidad del Cauca
Facultad de Ingeniería Electrónica y Telecomunicaciones
Departamento de Telemática
Popayán, Junio de 2006

TABLA DE CONTENIDO

1	PROYECTO E-LANE Y SISTEMAS DE GESTIÓN DE APRENDIZAJE.....	4
1.1	E-LEARNING	4
1.2	EL PROYECTO E-LANE EN LA UNIVERSIDAD DEL CAUCA	6
1.2.1	<i>EVA: Entorno Virtual de Aprendizaje para Unicauca.....</i>	<i>7</i>
1.3	SISTEMAS DE GESTIÓN DEL APRENDIZAJE (LMS) O PLATAFORMAS DE TELEFORMACIÓN	8
1.3.1	<i>Algunos de los Sistemas de Gestión de Aprendizaje más populares.....</i>	<i>9</i>
1.3.1.1	Moodle.....	9
1.3.1.2	ATutor	10
1.3.1.3	Dokeos	10
1.3.1.4	Claroline	10
1.3.2	<i>Framework de aplicaciones Web OpenACS</i>	<i>11</i>
1.3.2.1	Arquitectura de OpenACS	11
1.3.2.2	Sistema de Gestión de aprendizaje dotLRN.....	14
1.3.2.3	Arquitectura de dotLRN.....	15
1.3.3	<i>AOLServer.....</i>	<i>16</i>
1.3.4	<i>TCL (Tool Command Language).....</i>	<i>17</i>
1.3.5	<i>PostgreSQL.....</i>	<i>18</i>
1.4	CONSTRUCCIÓN Y GENERACIÓN DE CONTENIDOS EDUCATIVOS PERSONALIZADOS..	19
1.4.1	<i>Desarrollo de contenidos en base a Objetos de Aprendizaje</i>	<i>19</i>
1.4.2	<i>Consideraciones para el diseño de contenidos.....</i>	<i>20</i>
1.4.3	<i>Los perfiles de Usuario</i>	<i>21</i>
1.4.3.1	El Profiling	21
1.4.3.2	El Perfil.....	21
1.4.3.3	Personalización	22
1.4.3.4	Creación de Perfiles.....	23
1.4.3.5	Contenido del Perfil de Usuario.....	24
1.4.4	<i>Generación de Contenidos con docbook.....</i>	<i>24</i>
2	ESPECIFICACIONES Y HERRAMIENTAS PARA GENERACIÓN DE CONTENIDOS.....	27
2.1	LOS ESTANDARES EN EL PROCESO DE E-LEARNING.....	27
2.2	ELEMENTOS A ESTANDARIZAR EN UN SISTEMA DE E-LEARNING	28
2.3	ORGANIZACIONES IMPLICADAS EN EL PROCESO DE ESTANDARIZACIÓN	30
2.3.1	<i>IEEE-LTSC.....</i>	<i>30</i>
2.3.2	<i>IMS Global Learning Consortium.....</i>	<i>31</i>
2.3.3	<i>ADL.....</i>	<i>32</i>
2.4	MODELO DE REFERENCIA PARA OBJETOS DE CONTENIDO COMPARTIBLE – SCORM ...	32
2.4.1	<i>Modelo de agregación de contenidos (CAM)</i>	<i>34</i>
2.4.2	<i>Entorno de ejecución de SCORM (RTE).....</i>	<i>34</i>
2.5	HERRAMIENTAS PARA LA GENERACIÓN DE CONTENIDOS	35
2.5.1	<i>Butterfly.....</i>	<i>39</i>
2.5.2	<i>Xerlin</i>	<i>40</i>
2.5.3	<i>XML Pro v2.01.....</i>	<i>40</i>
2.5.4	<i>Borges.....</i>	<i>40</i>
2.5.5	<i>Exitor</i>	<i>41</i>
2.5.6	<i>Bitflux Editor.....</i>	<i>41</i>
2.5.7	<i>Vex - A Visual Editor for XML.....</i>	<i>42</i>
2.5.8	<i>Herramientas con TCL.....</i>	<i>42</i>
2.6	ANÁLISIS DE LAS HERRAMIENTAS ENCONTRADAS	42
3	GENERACIÓN DE CONTENIDOS Y RUTAS DE APRENDIZAJE PERSONALIZADOS.....	47

3.1	ELEMENTOS CLAVES DEL PERFIL DE UN ESTUDIANTE EN UN ENTORNO DE APRENDIZAJE MEDIADO POR TECNOLOGÍAS TELEMÁTICAS	47
3.1.1	<i>Estilo de aprendizaje</i>	47
3.1.2	<i>Tecnología de acceso del estudiante</i>	51
3.1.2.1	Identificación del navegador y el sistema operativo.....	52
3.1.2.2	Identificación de la velocidad de conexión.....	53
3.1.3	<i>Usabilidad de los contenidos</i>	55
3.2	ACCESIBILIDAD Y USABILIDAD EN EL DISEÑO DE CONTENIDOS PERSONALIZADOS... 58	
3.3	SISTEMAS RECOMENDADORES COLABORATIVOS PARA MATERIAL DE APRENDIZAJE MULTIMEDIA.....	60
3.4	DISEÑO DE OBJETOS Y ACTIVIDADES DE APRENDIZAJE [86] [87]	61
3.5	GUÍA METODOLÓGICA PARA LA CONSTRUCCIÓN DE CONTENIDOS Y RUTAS DE APRENDIZAJE ADAPTADOS AL PERFIL DEL ESTUDIANTE	64
3.5.1	<i>Lineamientos para la construcción de contenidos y rutas de aprendizaje personalizados</i>	65
3.5.2	<i>Lineamientos para la construcción de una herramienta Web con soporte para la generación de contenidos de aprendizaje</i>	70
4	ANÁLISIS DE UN ENTORNO INTEGRADO DE DESARROLLO PARA EL PROYECTO E-LANE EN LA UNIVERSIDAD DEL CAUCA	73
4.1	CONFIGURACIÓN DEL ENTORNO VIRTUAL DE APRENDIZAJE	73
4.2	SISTEMA RACE (RUTAS DE APRENDIZAJE SOBRE CONTENIDOS EDUCATIVOS)	77
4.2.1	<i>Arquitectura</i>	78
4.3	PROCESO DE DESARROLLO	80
4.3.1	<i>Modelo de la Organización</i>	81
4.3.2	<i>Modelo de Casos de Uso del Sistema</i>	84
4.3.3	<i>Modelo de Análisis del Sistema</i>	87
4.3.4	<i>Modelo de Diseño del Sistema</i>	96
4.3.5	<i>Modelo de Implementación del Sistema</i>	106
4.3.6	<i>Modelo de Despliegue del Sistema</i>	107
5	CASO DE ESTUDIO	108
5.1	CONSTRUCCIÓN DE UN PROTOTIPO DE CONTENIDO EDUCATIVO DIRIGIDO A ESTUDIANTES CON DIFERENTES PERFILES	108
5.2	ASPECTOS TECNOLÓGICOS CONSIDERADOS	121
5.3	ASPECTOS DE USABILIDAD CONSIDERADOS	121
	CONCLUSIONES, RECOMENDACIONES Y TRABAJOS FUTUROS	123

LISTA DE TABLAS Y FIGURAS

<i>Figura 1</i>	<i>Arquitectura del Framework OpenACS</i>	12
<i>Figura 2</i>	<i>Arquitectura del LMS dotLRN</i>	16
<i>Figura 3</i>	<i>Factores condicionantes del diseño de contenidos en tele-enseñanza</i>	21
<i>Figura 4</i>	<i>Clasificación de los estándares de la industria del e-learning</i>	29
<i>Figura 5</i>	<i>Estructura de un SCO [38]</i>	33
<i>Tabla 1</i>	<i>Descripción de los componentes de un SCO</i>	33
<i>Tabla 2</i>	<i>Exploración tecnológica para editores XML</i>	38
<i>Tabla 3</i>	<i>Características usadas como descarte de algunas herramientas Wysiwym</i>	39
<i>Tabla 4</i>	<i>Análisis comparativo de algunos editores XML</i>	43
<i>Tabla 5</i>	<i>Comparación de los Modelos de Estilos de Aprendizaje</i>	50
<i>Tabla 6</i>	<i>Clasificación de los estilos de aprendizaje de cada modelo</i>	51
<i>Figura 6</i>	<i>Estimación del uso de navegadores Web con respecto al tiempo</i>	52

Figura 7 Utilización del ancho de banda al inicio de una descarga	54
Tabla 7 Ejemplo de una Unidad Didáctica	63
Figura 8 Ejemplo de tareas comunes	63
Figura 9 Vista general del sistema y su entorno	78
Figura 10 Arquitectura del sistema a implementar	79
Tabla 8 Estructura de la Organización	82
Figura 11 Diagrama de Casos de Uso de la Organización	83
Figura 12 Diagrama de Casos de Uso del sistema RACE	84
Figura 13 Diagrama de paquetes de análisis esenciales RACE Editor	88
Figura 14 Diagrama de Clases de análisis Crear Book	88
Figura 15 Diagrama de Clases de análisis Adicionar Recurso	89
Figura 16 Diagrama de Clases de análisis Previsualizar Contenido	89
Figura 17 Diagrama de paquetes de análisis esenciales RACE Profile	90
Figura 18 Diagrama de Clases de análisis Identificar Estilo de Aprendizaje	91
Figura 19 Clases de análisis Identificar Tipo de Conexión	91
Figura 20 Clases de análisis Seleccionar Tipo de Conexión	92
Figura 21 Diagrama de paquetes de análisis esenciales RACE Content	92
Figura 22 Diagrama de Clases de análisis Recuperar Contenido	93
Figura 23 Diagrama de Clases de análisis Mapear Contenido y Agregar Estilo	93
Figura 24 Diagrama de Clases de análisis Generar Manifest	94
Figura 25 Diagrama de paquetes de análisis esenciales RACE Agent	94
Figura 26 Diagrama de Clases de Análisis Publicar Contenido	95
Figura 27 Diagrama de Clases de análisis Examinar Parámetros	95
Figura 28 Diagrama de Clases de análisis Generar Presentación	96
Figura 29 Diagrama de Clases de análisis Exportar Contenido	96
Figura 30 Estructura del sistema a implementar	97
Figura 31 Diagrama de Clases de Diseño Crear Book	99
Figura 32 Diagrama de Clases de Diseño Adicionar Recurso	99
Figura 33 Diagrama de Clases de Diseño Previsualizar Contenido	100
Figura 34 Diagrama de Clases de Diseño Identificar Estilo de Aprendizaje	101
Figura 35 Diagrama de Clases de Diseño Identificar Tipo de Conexión	101
Figura 36 Clases de diseño caso de uso Seleccionar Tipo de Conexión	102
Figura 37 Diagrama de Clases de Diseño Recuperar Contenido	102
Figura 38 Diagrama de Clases de Diseño Mapear Contenido y Agregar Estilo	103
Figura 39 Diagrama de Clases de Diseño Generar Manifest	103
Figura 40 Diagrama de Clases de Diseño Publicar Contenido	104
Figura 41 Diagrama de Clases de Diseño Examinar Parámetros	104
Figura 42 Diagrama de Clases de Diseño Generar Presentación	105
Figura 43 Diagrama de Clases de Diseño Exportar Contenido	105
Figura 44 Diagrama de Implementación del Sistema	106
Figura 45 Diagrama de Despliegue del Sistema	107
Figura 46 Estructura del Contenido Desarrollado	110
Figura 47 Distribución espacial para el desarrollo del contenido	111
Figura 48 Selección del estilo de aprendizaje y el tipo de conectividad	112
Figura 49 Ejemplo de los objetivos definidos para una unidad didáctica	113
Figura 50 Herramienta de modelado a nivel Web: gmodeler [95]	115
Figura 51 Contenido para el estudiante asimilador + conectividad baja	116
Figura 52 Contenido para el estudiante convergente + conectividad baja	117
Figura 53 Contenido para el estudiante convergente + conectividad alta	118
Figura 54 Revisión adicionada al contenido	119
Figura 55 Publicación del contenido en LORS	120

INTRODUCCIÓN

En muchas instituciones educativas, principalmente en las instituciones de educación superior, se está utilizando la modalidad de e-learning como mecanismo de apoyo a las prácticas convencionales de formación académica. La utilización de Internet como soporte de comunicación e interacción entre el profesor y los estudiantes complementa adecuadamente los procesos formativos. La calidad de los sistemas actuales de e-learning es buena, pero cada día se hace necesario desarrollar nuevas herramientas al interior de estos sistemas que satisfagan las necesidades de los actores del proceso educativo con el fin de obtener mejores resultados.

El presente proyecto realiza un aporte al mejoramiento de la calidad del entorno virtual de aprendizaje dotLRN, un Sistema de Gestión de Cursos y Comunidades de código abierto escogido por el proyecto E-LANE, como plataforma de soporte para la realización de cursos en modalidad virtual, debido a sus características de escalabilidad, robustez, extensibilidad y cumplimiento de estándares. Las contribuciones realizadas están enmarcadas en los conceptos de adaptabilidad y usabilidad, aunque no se dejó de lado el concepto de accesibilidad, no se pudo aplicar como hubiera sido ideal, estos conceptos constituyen las características que deberían tener las herramientas informáticas educativas y en especial los sistemas que posibilitan el aprendizaje a través de Internet. Un sistema adaptable es aquel que permite al usuario cambiar ciertos parámetros del sistema y adaptar su comportamiento de acuerdo a sus necesidades. Un sistema usable es aquel que puede ser utilizado por los usuarios para alcanzar unos objetivos específicos al realizar una tarea con efectividad. Un sistema accesible es el que se acomoda, principalmente desde el punto de vista de las interfaces de usuario, a las diferencias de las personas permitiéndoles usarlo sin problemas.

Actualmente los sistemas educativos virtuales tratan de imitar las formas tradicionales de docencia del profesor y no consideran en algunos aspectos al aprendiz, por eso resulta importante no solo el desarrollo de contenidos de calidad bajo recomendaciones pedagógicas y tecnológicas sino la forma de aprender del estudiante (estilo de aprendizaje). El proyecto se enfoca hacia dichos aspectos, es decir, tiene en cuenta la forma en que el

estudiante aprende o su estilo de aprendizaje, su velocidad de conexión y el desarrollo de contenidos de calidad bajo lineamientos pedagógicos fundamentales.

La necesidad de adaptación de los contenidos surge a partir de la pregunta ¿es la información presentada, la adecuada para el estudiante? Mediante la aplicación de instrumentos pedagógicos como los estilos de aprendizaje en los sistemas de e-learning se consigue un aprendizaje más efectivo, instrumento que no es tan fácil de aplicar en las clases presenciales. La personalización de los contenidos adaptándose a los estilos de aprendizaje de cada estudiante favorece la interacción estudiante-contenidos lo cual mejora el proceso de comprensión de la información y motiva al estudiante en el desarrollo de las actividades académicas potencializando de esta manera el aprendizaje.

Además de los factores propios del estudiante como el estilo de aprendizaje, otros aspectos como la capacidad de conexión a Internet de la cual dispone el estudiante son importantes en el proceso de adaptabilidad, puesto que los contenidos de aprendizaje varían en tamaño y en algunos casos pueden ser muy “pesados” para ser accedidos por el estudiante desde su conexión actual a Internet. Por esto se hace necesario, manejar contenidos acordes al tamaño o peso de los contenidos. Desde este punto de vista, es necesario realizar una presentación adaptada de los contenidos para poder garantizar el acceso de todos los estudiantes a la información del contenido.

El sistema desarrollado está compuesto por cuatro subsistemas. En primer lugar, una herramienta para la generación de contenidos educativos personalizados de acuerdo al estilo de aprendizaje del estudiante y su velocidad de conexión. El nombre de esta herramienta es RACE Editor¹ y constituye un editor muy intuitivo utilizado por el profesor, considerando el estilo de aprendizaje y la velocidad de conexión a Internet de la que dispone el estudiante, para la creación, edición y publicación de contenidos en el repositorio de objetos de aprendizaje (LORS) de dotLRN. En segundo lugar, una aplicación utilizada por el estudiante para el establecimiento de su perfil, es decir, para conocer su estilo de aprendizaje y la velocidad de su conexión al utilizar el entorno virtual de aprendizaje dotLRN. El nombre de

¹ **RACE Editor:** Herramienta para la generación de contenidos educativos del sistema para la definición de rutas de aprendizaje sobre contenidos educativos (RACE).

esta herramienta es RACE Profile². En tercer lugar, un subsistema para la personalización de los contenidos creados mediante RACE Editor de acuerdo al perfil del estudiante establecido mediante RACE Profile, este subsistema esta soportado tanto por el mismo RACE Editor en la parte de creación de contenidos personalizados como por un servicio llamado RACE Agent³. En cuarto lugar se tiene un servicio llamado RACE Content⁴, el cual posee los métodos adecuados para el procesamiento y entrega de contenidos educativos. Este sistema se basa en la creación de una presentación adaptada de los contenidos al estilo de aprendizaje y a la velocidad de conexión del estudiante, según el tipo de recursos (texto, imágenes, audio, video, etc.) a presentarse para cada perfil.

Adicionalmente al involucrar algunos aspectos relacionados con la usabilidad, se deja una puerta abierta para explorar en profundidad este amplio campo de conocimiento, y utilizarlo en la creación de contenidos educativos de mayor calidad.

En cuanto al presente documento, está organizado de la siguiente forma:

El Capítulo 1, presenta los conceptos teóricos acerca del Framework de aplicaciones Web OpenACS y el Sistema de gestión de aprendizaje dotLRN sobre los cuales se desarrolló RACE. También incluye los aspectos relevantes en la construcción y generación de contenidos educativos personalizados. El capítulo 2, incluye información sobre el proceso de búsqueda de una herramienta que pueda ofrecer soporte a la generación de contenidos orientados a ser utilizados en entornos virtuales de aprendizaje tales como dotLRN, al igual que los mecanismos de estandarización para la creación y utilización de recursos educativos en dichos entornos. El Capítulo 3, incluye la descripción de las características del perfil del estudiante al igual que algunos lineamientos fundamentales para la creación de una herramienta a nivel Web para desarrollo de contenidos educativos con soporte a personalización, este capítulo incluye también la formulación de una guía metodológica para la construcción de contenidos adaptados al perfil del estudiante como también algunos lineamientos a nivel tecnológico, específicamente a nivel de velocidad de conexión,

² **RACE Profile:** Es un sistema identificador de la velocidad de conexión y el estilo de aprendizaje del estudiante en el sistema RACE.

³ **RACE Agent:** Subsistema de control de RACE.

⁴ **RACE Content:** Subsistema con métodos adecuados para manipulación y transformación de contenidos basados en XML de RACE.

finalmente se incluye también algunos lineamientos relacionados con la usabilidad. El capítulo 4, contiene una descripción de algunas herramientas que forman parte del entorno integrado de desarrollo y pueden brindar soporte al proceso de construcción de contenidos, tareas o actividades de aprendizaje personalizadas basadas en los diferentes lineamientos que aparecen en el capítulo 3. De igual forma, incluye el proceso de diseño, implementación e integración del sistema RACE. El Capítulo 5 describe el proceso de construcción de un prototipo de una parte de un curso dirigido a estudiantes con diferentes perfiles y diferente conectividad aplicando la guía metodológica formulada en el capítulo 3. Finalmente se exponen las conclusiones a las cuales se llegó en la etapa de finalización del proyecto, brindando las recomendaciones respectivas y planteando los trabajos relacionados que podrían desarrollarse en un futuro.

1 PROYECTO E-LANE Y SISTEMAS DE GESTIÓN DE APRENDIZAJE

1.1 E-LEARNING

Las tecnologías de la información y las comunicaciones (TIC) posibilitan la creación de un nuevo espacio social-virtual para las relaciones humanas, este nuevo entorno se está desarrollando en el área de la educación ofreciendo una buena alternativa a los procesos tradicionales de enseñanza y aprendizaje.

Las redes telemáticas constituyen las nuevas unidades del sistema educativo que conllevan al diseño y la construcción de nuevos escenarios educativos y a la elaboración de materiales educativos electrónicos para ser distribuidos principalmente a través de Internet, el cual se convierte en la herramienta más importante de soporte al proceso educativo [1].

El e-learning es un nuevo concepto de educación a distancia soportada en el uso de las TIC y materiales en formato digital como documentos y elementos multimedia. Posibilita el desarrollo de las actividades relativas al proceso educativo tales como la ejecución de un curso académico, incluyendo sistemas de evaluación del estudiante y de seguimiento de su

proceso de aprendizaje que permiten mejorar la calidad del proceso. Ofrece herramientas y servicios que favorecen la comunicación entre los diferentes actores del proceso, principalmente entre el estudiante y el profesor y entre los estudiantes. Entre los más comunes se encuentran el e-mail, los foros y el chat que posibilitan un proceso educativo más interactivo.

En la actualidad, la mayor parte de las instituciones de educación superior está utilizando la modalidad del e-learning para brindar soporte a las actividades formativas, utilizando una diversa gama de plataformas software conocidas como LMS (Learning Management Systems) o *Sistemas de Gestión del Aprendizaje*. Estos sistemas permiten realizar diferentes procesos como la gestión y el control de los cursos, la gestión de los usuarios y la gestión de los servicios de comunicación entre los miembros [2].

En el proceso de e-learning hay cuatro factores importantes a considerar, la plataforma de estudio (dotLRN para el caso de este proyecto), el usuario para quien se define diferentes roles como el de estudiante, profesor y administrador del sistema, los contenidos y la modalidad de enseñanza. El entorno virtual es el equivalente al aula física tradicional y es el espacio al cual el estudiante accede para tomar las lecciones y contactarse con el profesor y con los otros estudiantes. Generalmente estas plataformas tienen la gran ventaja de que pueden configurarse, modificarse e incluso mejorarse según las necesidades de los usuarios ofreciendo cada vez mejores servicios.

En los LMS se utiliza muy a menudo los conceptos de profesor, tutor y estudiante. El profesor es quien crea los contenidos que en conjunto constituyen un curso y los gestiona continuamente según sus requerimientos y los de los estudiantes. El tutor es un colaborador del estudiante y se encarga de guiarlo en el desarrollo de las actividades de aprendizaje al interior del LMS y de responder algunas de sus inquietudes, es muy común que las funciones de profesor y tutor sean llevadas a cabo por la misma persona. El estudiante o aprendiz es sobre quien recae el proceso de formación virtual, accede a los contenidos de un curso y los utiliza en su proceso de apropiación de la información. El administrador del sistema es la persona que se encarga de las tareas generales de gestión de la plataforma como la asignación de diferentes niveles de acceso al sistema o a los cursos según el rol definido para cada usuario.

Los contenidos utilizados varían de acuerdo con los requerimientos específicos de cada estudiante o grupo de estudiantes matriculados en un determinado curso, dichos contenidos pueden tener aspecto variado, desde documentos en texto plano que el aprendiz tiene la posibilidad de descargar o documentos en formato de hipertexto que pueda visualizar en la plataforma por medio de un navegador Web hasta aplicaciones multimedia para las cuales el usuario debe utilizar visualizadores específicos. Lo importante de los contenidos es que estén bien diseñados, estructurados, redactados y se puedan gestionar fácilmente [3].

En cuanto a la enseñanza existen dos modalidades, la sincrónica y la asincrónica. En la primera los educandos tienen que acudir al aula virtual a determinada hora para recibir la información que el profesor entrega y en algunas ocasiones interactuar con él dependiendo del tipo de contenido (videoconferencia, audioconferencia, etc.), pero se presentan problemas de funcionamiento en especial para los usuarios que no disponen de un ancho de banda suficiente. En la asincrónica los contenidos están disponibles para que el educando pueda tener acceso a ellos en cualquier momento y por lo tanto no es necesaria la presencia del profesor, tampoco requiere de conexiones con capacidades muy grandes pues la comunicación estudiante-profesor no se realiza en tiempo real. Es por eso que la mayoría de las propuestas de e-learning ofrecen cursos asincrónicos. Algunas instituciones educativas ofrecen cursos en ambas modalidades a elección del estudiante.

1.2 EL PROYECTO E-LANE EN LA UNIVERSIDAD DEL CAUCA

E-LANE (Nueva Educación para Europa y Latinoamérica) es un proyecto que tiene por cobertura muchas de las universidades de Europa y América Latina y tiene como objetivo producir material educativo de alta calidad y a bajo costo mediante la modalidad de e-learning, basándose en la utilización de una plataforma de software libre y en el diseño de una metodología de enseñanza innovadora orientada hacia esta plataforma. El proyecto E-LANE pretende generar un gran impacto a nivel de formación académica y promover un entorno sólido para la alfabetización digital y el aprendizaje en la sociedad de la información [4].

Para este trabajo de grado son fundamentales tres de los objetivos específicos del proyecto E-LANE, los cuales son:

- *La plataforma de formación.* Integración del Estado del Arte de las tecnologías de la información y las comunicaciones relacionadas con la teleformación para obtener una plataforma de programación abierta, con mínimos requisitos de equipamiento, que suministre la funcionalidad esencial para orientar cursos en diferentes disciplinas y escenarios (presencial vs. no presencial, síncrono vs. asíncrono).
- *Metodología Innovadora.* De igual importancia que la plataforma de programación es la construcción de una metodología que saque provecho de los nuevos recursos dispuestos por la tecnología. Más concretamente, tanto la manera de orientar los cursos como la forma de preparar el material deben tener en consideración el uso de un entorno de teleformación.
- *Contenidos de los cursos.* Producir cursos para ser orientados con las herramientas y metodologías propuestas. Se dispondrá de material para implementar cursos apropiados que resuelvan las necesidades en educación de las sociedades latinoamericanas. Un aspecto que es crucial para la elaboración del material es la adaptación de la infraestructura requerida para que sea capaz de soportar un alto grado de reutilización [5].

1.2.1 EVA: Entorno Virtual de Aprendizaje para Unicauca

En la Universidad del Cauca se ha implantado la plataforma de enseñanza-aprendizaje EVA (Entorno Virtual de Aprendizaje) que es una adaptación del Sistema de Gestión de Aprendizaje dotLRN⁵. A través de esta plataforma se está ofreciendo soporte a la impartición de cursos presenciales y se están probando escenarios donde es utilizada como medio para ofrecer cursos en modalidad virtual. A la plataforma se han ido incorporando mejoras progresivas aprovechando la experiencia que tienen algunos miembros de la Universidad

⁵ dotLRN: Sistema de Gestión del Aprendizaje desarrollado en el MIT (Instituto Tecnológico de Massachussets).

Galileo (Guatemala) y las contribuciones realizadas por la Universidad Carlos III de Madrid (España) [6].

1.3 SISTEMAS DE GESTIÓN DEL APRENDIZAJE (LMS) O PLATAFORMAS DE TELEFORMACIÓN

Actualmente el concepto de LMS está experimentando una migración hacia el nuevo concepto de LCMS (Sistema de Gestión de Contenidos de Aprendizaje) debido al desarrollo e integración al LMS de aplicaciones que posibilitan la generación y gestión de contenidos educativos en formatos como XML y HTML. Por lo tanto, al profesor se le ofrece una alternativa para producir los contenidos de un curso mediante una herramienta incluida en la plataforma de teleformación, lo que elimina la necesidad de utilizar programas de escritorio tales como las herramientas de oficina (Office, OpenOffice, etc.).

Los LCMS ofrecen herramientas para el desarrollo flexible y la entrega de contenidos y posibilitan procesos de seguimiento y trazabilidad, control del aprendizaje, reusabilidad de los contenidos, funciones de administración, seguridad, etc. Los LMS y en especial los LCMS poseen las siguientes características:

- Posibilitan el acceso remoto tanto a profesores como a estudiantes en cualquier momento y desde cualquier lugar por medio de una conexión a Internet o a redes que utilicen el protocolo TCP/IP.
- Utilizan un navegador. Permiten a los usuarios acceder a la información a través de navegadores estándares como Netscape, Internet Explorer, Opera y Mozilla utilizando el protocolo de comunicación http.
- El acceso es independiente del sistema operativo y del computador de cada usuario. Es decir, utilizan estándares de manera que la información pueda ser visualizada y tratada en las mismas condiciones, con las mismas funciones y con el mismo aspecto en cualquier computador.
- Tienen una estructura Cliente-Servidor.
- El acceso a la plataforma es restringido y selectivo.

- Incluyen como elemento básico una interfaz gráfica común, con un único punto de acceso de manera que en ella se integran los diferentes elementos multimedia que constituyen los cursos: texto, gráficos, imágenes, animaciones, sonidos y video.
- Utilizan páginas elaboradas con los estándares HTML y XML aceptados por el protocolo http.
- Realizan la presentación de la información en formato multimedia. Los formatos HTML y XML permiten presentar la información de forma adecuada, además de hipertexto pueden utilizarse gráficos, animaciones, audio y video.
- Permiten al usuario acceder a recursos y a cualquier información disponible en Internet a través de enlaces o a través del propio entorno de la plataforma.
- Permiten estructurar la información en formato hipertextual. De esta manera la información se puede organizar, estructurada a través de enlaces y asociaciones de tipo conceptual y funcional de tal forma que esto sea perceptible por los usuarios.
- Permiten establecer diferentes roles de usuario con distintos privilegios de acceso. Debe contemplar al menos: el administrador, que se encarga del mantenimiento del servidor y de administrar claves y privilegios; el coordinador o responsable de un curso, es el perfil del profesor que diseña y se responsabiliza del desarrollo del curso y de la coordinación docente y organizativa del curso en la plataforma; los tutores, encargados de la atención de los estudiantes y de la elaboración de materiales educativos y por último los aprendices o estudiantes. [7]

1.3.1 Algunos de los Sistemas de Gestión de Aprendizaje más populares

A continuación se describe brevemente algunos de los Sistemas de Gestión de Aprendizaje de libre distribución mas utilizados actualmente.

1.3.1.1 Moodle

Moodle es un paquete de software para la creación de cursos y sitios Web basados en Internet. Es un proyecto en desarrollo diseñado para dar soporte a un marco de educación social constructivista. Se distribuye gratuitamente como Software libre (Open Source) bajo la licencia pública GNU. Moodle puede funcionar en cualquier computador en el que pueda

correr PHP y soporta varios tipos de bases de datos (en especial MySQL). La palabra Moodle era al principio un acrónimo de Modular Object-Oriented Dynamic Learning Environment (Entorno de Aprendizaje Dinámico Modular y Orientado a Objetos). [8]

1.3.1.2 ATutor

Es un Sistema de Gestión de Contenidos de Aprendizaje (LCMS) de Código Abierto basado en la Web y diseñado con el objetivo de lograr accesibilidad y adaptabilidad. Los administradores pueden instalar o actualizar ATutor en minutos. Los educadores pueden rápidamente ensamblar, empaquetar y redistribuir contenido educativo y llevar a cabo sus clases en línea. Los estudiantes pueden aprender en un entorno de aprendizaje adaptativo. ATutor cumple con las especificaciones W3C WCAG 1.0 accessibility, posibilitando el acceso a todos los estudiantes, instructores y administradores. ATutor ha adoptado también las especificaciones IMS/SCORM Content Packaging, permitiendo a los desarrolladores de contenido crear contenido reutilizable que puede ser intercambiado entre diferentes sistemas e-learning. [9]

1.3.1.3 Dokeos

Es una aplicación Web de código abierto para gestión de cursos y colaboración en Internet. Esta permite al entrenador crear contenido pedagógico, estructurar actividades en rutas de aprendizaje, interactuar con los estudiantes y seguir su evolución a través de un sistema de informes o reportes. Sus características para gestión de cursos incluyen distribución de contenido, calendario, seguimiento del progreso del estudiante, texto, audio, video, chat, administración de evaluaciones y actualización de registros. [10]

1.3.1.4 Claroline

Es una aplicación libre basada en PHP/MySQL. Constituye un ambiente de aprendizaje colaborativo que le permite a los profesores o a las instituciones educativas crear y administrar cursos a través de la Web. El sistema provee gestión de grupos, foros, repositorios de documentos, calendario, chat, asignación de áreas, enlaces, administración del perfil de usuario en un paquete simple y altamente integrado. [11]

1.3.2 Framework⁶ de aplicaciones Web OpenACS

OpenACS (Open Architecture Community System) es un conjunto de aplicaciones software de código abierto muy potente para el desarrollo de aplicaciones Web. Fue desarrollado inicialmente por una empresa llamada ArsDigita y posteriormente liberado bajo licencia GPL [12]. Ha demostrado ser una herramienta muy robusta capaz de brindar solución a proyectos de gran envergadura. Se constituye como una aplicación Framework proporcionando diferentes beneficios a los desarrolladores de nuevas aplicaciones como modularidad, reusabilidad y extensibilidad. OpenACS permite a los diseñadores de sitios Web producir aplicaciones modulares y rápidamente extensibles y a la vez reutilizar gran cantidad de software empleando las características de los paquetes que forman el núcleo de la aplicación.

1.3.2.1 Arquitectura de OpenACS

El Framework OpenACS está programado principalmente en TCL (Tool Command Language) y está constituido por un conjunto de aplicaciones que actúan entrelazadamente con el servidor Web AOLserver y la base de datos PostgreSQL. Está dividido en varios paquetes, unos pertenecientes al Core o núcleo del sistema, otros a los servicios y otros a las aplicaciones ofrecidas por el Framework. Los paquetes del Core son aquellos que se requieren para que OpenACS funcione, estos incluyen un Kernel, gestión de paquetes, utilidades para la administración de los usuarios, un sitemap, un sistema de plantillas que separa claramente la lógica de negocio de la lógica de presentación y un sistema de mensajería y correo. Los paquetes servicio tales como el paquete de eventos son aplicaciones de lógica reutilizable, independientes y no implementan interfaces de usuario. Los paquetes que se denominan aplicación tienen una interfaz de usuario y su código está claramente dividido en la lógica de aplicación y la lógica de presentación. La arquitectura de OpenACS se observa en la figura 1.

⁶ **Framework:** Puede definirse como un conjunto de aplicaciones software reutilizables que pueden emplearse para producir aplicaciones personalizadas.

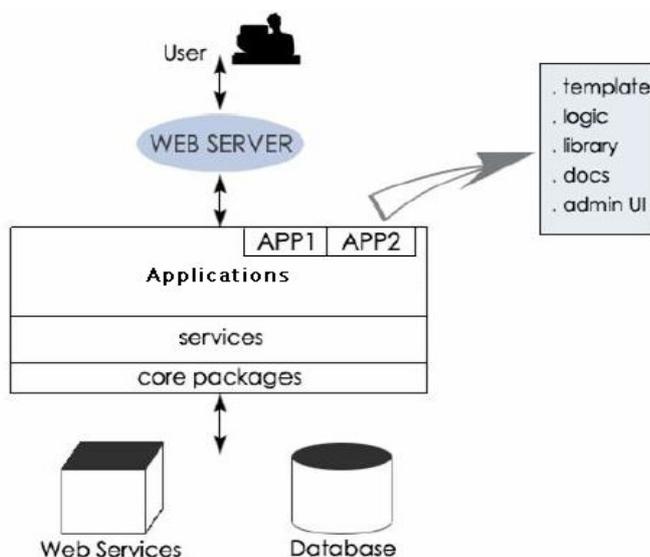


Figura 1 Arquitectura del Framework OpenACS

Cada servicio y aplicación requieren un modelo de datos que debe ser integrado en el Framework. La integración de estos modelos de datos permite la reusabilidad y diferentes funcionalidades. Para aumentar la reusabilidad, el Framework tiene una arquitectura orientada a objetos pero, ya que el RDBMS (PostgreSQL) no implementa las funcionalidades de programación orientada a objetos, estos deben ser añadidos intrínsecamente por el Framework.

Los objetos de OpenACS tienen sus atributos guardados en tablas y un juego de métodos definidos como los paquetes del API PL/SQL. Estos paquetes contienen los procedimientos que proporcionan la interface para el modelo de datos. Las estructuras de los datos se integran en el Framework creando una tabla llamada "acs_objects" que guarda un registro de cada objeto de OpenACS. Otra tabla llamada "acs_object" define los atributos generales guardados en cada objeto, incluyendo un identificador ID único que representa el tipo de objeto. El concepto de los tipos de objetos en OpenACS es equivalente a las clases en los lenguajes de programación orientados a objetos.

Los tipos de objeto se especifican mediante atributos definidos en las tablas acs_object_types y acs_attributes. Los tipos del objeto en OpenACS pueden heredar los atributos de un tipo del padre, por lo tanto los tipos del sistema son jerárquicos. Para utilizar

el sistema de objetos de OpenACS se requieren cuatro pasos, los cuales se llevan a cabo con el modelo de datos de la aplicación: Crear el tipo del objeto, crear los atributos del objeto, crear los métodos del objeto y crear la tabla para almacenar los datos del objeto.

Hay 3 aspectos importantes en OpenACS que son, el proceso de peticiones, el sistema de permisos y el sistema de plantillas.

- *Proceso de las peticiones:* Las peticiones de una página en OpenACS son manejadas por un procesador de peticiones (RP) que se encarga de recibir la solicitud, autenticar al usuario, examinar los permisos del usuario y de acuerdo a eso procesar la solicitud desplegando la información solicitada.
- *Sistema de permisos:* El sistema de permisos hace posible que los diseñadores y administradores del sistema puedan definir lo que se permite hacer a los usuarios. Se establece control de acceso a nivel del objeto, es decir, cualquier aplicación u objeto del sistema representado por una fila en la tabla `acs_objects` puede controlarse mediante una simple sentencia PL/SQL. Cada objeto tiene un atributo llamado "context_id" que especifica los permisos predefinidos para un objeto y su alcance.
- *Sistema de plantillas:* OpenACS utiliza un sistema de plantillas para separar de forma clara y en diferentes conjuntos de archivos la lógica del negocio y la lógica de presentación en las aplicaciones. Utiliza un subdirectorío estándar (el `www`) donde cada usuario que visualiza una página se almacena en por lo menos dos archivos distintos. Los archivos `.tcl` tienen estructuras de programación de acuerdo al lenguaje `tcl` y sentencias SQL que actúan como las fuentes de datos. Los ficheros `.adp` devuelven los valores entregados desde los archivos `.tcl` en HTML usando un juego de etiquetas especiales y sustituciones de esas fuentes de datos. Muchas aplicaciones requieren lógica de administración la cual se separa del resto en el directorio `admin`. Una aplicación de OpenACS también requiere documentación almacenada en otro directorio (el `doc`). El HTML archivado ahí es manejado por un sistema de documentación centralizado. Finalmente, un paquete completo puede tratarse como otro objeto en OpenACS, lo cual significa que pueden instalarse distintas instancias del mismo paquete en el mismo sitio web. [13]

1.3.2.2 Sistema de Gestión de aprendizaje dotLRN

dotLRN es un sistema de e-learning y gestión de comunidades⁷ que está basado en el Framework OpenACS. Es un proyecto de código abierto, escalable, robusto y extensible que proporciona gestión de cursos y ofrece un conjunto de aplicaciones colaborativas y servicios basados en estándares como SCORM⁸. La arquitectura fue diseñada por ingenieros del MIT⁹ para facilitar la implementación de comunidades virtuales con un mínimo de esfuerzo, y al mismo tiempo con la suficiente capacidad de extenderse a nuevas áreas de desarrollo, incluso a aquellas que no fueron contempladas en el diseño original. [14]

dotLRN está construido sobre el Framework de aplicaciones Web OpenACS, el cual implementa un modelo de datos orientado a objetos que los programadores pueden modificar. Los usuarios de dotLRN tienen una interfaz Web que les permite crear departamentos (por ejemplo, el departamento de Telemática) materias y clases dentro de las cuales se distribuyen los cursos. Cada uno de estos cursos tiene una página creada automáticamente (portal) dentro de la cual aparecen todas las aplicaciones que el profesor considere importante para el curso. La apariencia de este portal puede ser adaptada por el profesor de forma sencilla desde la misma página. Cada usuario del sistema (Profesor, Estudiante, Administrador, etc.) tiene una página personal donde la información de todos los cursos y comunidades a las que pertenece es compilada y presentada adecuadamente.

Cada portal incluye ciertas aplicaciones como Foros, que permiten establecer discusiones sobre un determinado tema entre el profesor y los estudiantes, Noticias, donde el profesor puede anunciar o recordar a los estudiantes eventos tales como exámenes y tareas y los Materiales de Aprendizaje, que conforman el contenido del curso. Además, es posible personalizar el portal adicionando una nueva aplicación o quitando alguna de las existentes según las necesidades de los usuarios, esto se realiza a través de mini ventanas llamadas Portlets que constituyen la vista de la aplicación y pueden ser habilitadas o deshabilitadas. dotLRN es el encargado de incluir la funcionalidad al portal en la parte de e-learning, gestión de estudiantes, clases, compartir archivos, etc.

⁷ Se entiende por comunidad un grupo de personas que comparten ciertos aspectos o pertenecen a un mismo dominio, por ejemplo, la comunidad educativa de la Universidad del Cauca.

⁸ **SCORM:** Modelo de Referencia de Objetos de Contenido Compartible

⁹ **MIT:** Instituto Tecnológico de Massachussets.

1.3.2.3 Arquitectura de dotLRN

La arquitectura de dotLRN básicamente es la misma del Framework OpenACS ya que dotLRN constituye un conjunto de paquetes que utilizan dicho Framework para funcionar. Esta arquitectura se muestra en la figura 2. El diagrama tiene cinco componentes principales: el usuario, el servidor Web, el Framework o armazón, la base de datos y los servicios. Los paquetes del ACS-core mantienen a través del API la funcionalidad común entre otros servicios y paquetes. Los servicios son paquetes que no interactúan con el usuario y establecen una interface entre los paquetes donde reside la aplicación real. Como dotLRN está construido con el Framework de OpenACS, hereda todas las funcionalidades de éste, su modularidad y características, por ejemplo, dotLRN usa el calendario de OpenACS. Distintas instancias del calendario pueden visualizarse en el portal del estudiante, mostrando en una sola página todos los eventos por cada curso en los que está matriculado. El sistema adapta el portal a cada usuario recogiendo la información sobre los cursos y comunidades a los cuales pertenece y las aplicaciones que utiliza. Como se mencionó en el párrafo anterior, el acceso a cada una de estas aplicaciones se produce a través de un portlet que se encarga de generar la interfaz apropiada entre las aplicaciones de OpenACS y los portales de dotLRN, de esta forma los portlets entregan funcionalidad personalizada basada en las aplicaciones de OpenACS. [15]

En dotLRN los usuarios pueden tener más de un rol en particular, por ejemplo, un usuario puede ser un instructor en una clase y un estudiante en otra. Por lo tanto, el papel de un usuario en una clase determina lo que al usuario se le permite hacer. Todos los nombres de roles y entidades en dotLRN, como la comunidad y la clase pueden cambiarse fácilmente de acuerdo a las necesidades de la institución donde el sistema se esté usando, tal es el caso de EVA, versión de dotLRN adaptada para su uso en la Universidad del Cauca.

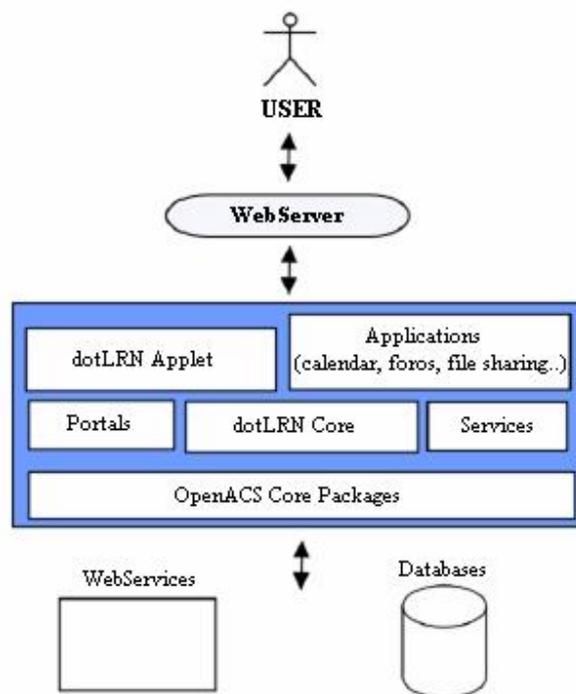


Figura 2 Arquitectura del LMS dotLRN

A continuación se presenta una descripción de los componentes de la arquitectura anterior que son externos a la plataforma dotLRN/OpenACS.

1.3.3 AOLServer

La plataforma dotLRN/OpenACS utiliza un servidor Web de código libre que se llama AOLServer. Fue desarrollado por AOL (América Online), el proveedor de Internet más importante del mundo. Es un servidor Web multihilo basado en el lenguaje TCL y con muchas facilidades de uso en entornos de gran escala y sitios Web dinámicos. AOLServer tiene una amplia base de usuarios gracias sobre todo a su integración con OpenACS. AOLServer está basado en páginas ADP (AOL Dynamic Server Pages) las cuales tienen el mismo funcionamiento que las ASP (Active Server Pages) de Microsoft y las JSP (Java Server Pages) de Sun. Básicamente son páginas HTML que utilizan o contienen fragmentos de código TCL. Una de las ventajas que posee AOLServer es que incluye un intérprete de

TCL embebido en su estructura, debido a esto es posible mantener Scripts en TCL que se ejecutarán internamente a diferencia de Apache, en donde al ejecutarse un Script CGI¹⁰ existe la necesidad de iniciar un nuevo proceso. AOLServer también tiene soporte para CGI, pero debido a la ejecución superior de Scripts internos en TCL, el uso de CGI no es tan amplio. AOLServer mantiene grupos de conexiones latentes o *pools* hacia bases de datos, estas conexiones permiten al servidor realizar búsquedas más rápidas (no es necesario abrir y cerrar continuamente la conexión con la base de datos), actualmente existen Drivers para las siguientes bases de datos: SOLID, PostgreSQL, Oracle, Sybase, Informix, Internase y MySQL [16]. Ya que AOLserver está escrito en TCL, proporciona varias funciones en TCL (un API) que permiten interactuar con la conexión de http, mantener permisos de URL y acceder bases de datos. AOLServer también proporciona un API en C en caso que se requiera escribir un módulo especial que realice control de acceso, drivers de comunicación u otras funcionalidades. [17]

1.3.4 TCL (Tool Command Language)

Es un lenguaje de programación interpretado y multiplataforma creado por John K. Ousterhout y su equipo de la Universidad de California pero actualmente es desarrollado por Sun Microsystems Laboratories (en concreto por el grupo SunScript que lidera el propio Ousterhout) y distribuido de forma totalmente gratuita. TCL es un lenguaje de comandos cuyo intérprete recibe el nombre de tclsh (tclsh80 para Tcl/Tk 8.0), que tiene como una de sus principales características la gran facilidad con la que se pueden implementar funciones en C/C++ que pasan a ser nuevas instrucciones del intérprete. Es decir, se pueden embeber aplicaciones en C/C++ dentro del propio intérprete de Tcl; de esta forma es posible obtener nuevas versiones de Tcl, denominadas *extensiones*, que no dejan de ser otra cosa que intérpretes que adicionan a los comandos originales de Tcl nuevos comandos escritos en C/C++.

Algunas de estas extensiones son BLT (que permite hacer representaciones gráficas en 2D), Itcl (Incremental Tcl, Tcl orientado a objetos), OraTcl (Tcl capaz de manejar bases de datos

¹⁰ **CGI:** Es una tecnología que permite a un cliente y desde su navegador solicitar datos de un programa ejecutado en el Servidor Web. CGI especifica un estándar para el paso de datos entre el cliente y el programa.

ORACLE), etc. Pero sin duda, la extensión más conocida y que es distribuida junto con el propio Tcl es Tk (Tool Kit). Tk, creada por el propio John Ousterhout, proporciona un intérprete denominado *wish* que añade a los comandos de Tcl comandos capaces de crear interfaces gráficas de usuario. Es decir, Tk permite crear ventanas, botones, menús, barras de scroll y toda una serie de elementos propios de la programación con ventanas. A todos estos elementos los denomina *widgets*. Tk se distribuye junto con Tcl en un paquete denominado Tcl/Tk, que proporciona los dos intérpretes citados anteriormente. El intérprete de Tcl puede ser eliminado ya que se encuentra incorporado en el intérprete de Tk. [18]

1.3.5 PostgreSQL

PostgreSQL es uno de los gestores de bases de datos objeto-relacionales más avanzados hoy en día, está liberado bajo la licencia BSD, lo que significa que cualquiera puede disponer de su código fuente, modificarlo a voluntad y redistribuirlo libremente. Postgres, desarrollada originalmente en el Departamento de Ciencias de la Computación de la Universidad de California en Berkeley, fue pionera en muchos de los conceptos de bases de datos relacionales orientadas a objetos que ahora empiezan a estar disponibles en algunas bases de datos comerciales. Soporta casi toda la sintaxis SQL (incluyendo subconsultas, transacciones y funciones definidas por el usuario), ofrece integridad de transacciones, extensibilidad de tipos de datos y control de concurrencia multi-versión y cuenta también con un amplio conjunto de enlaces con lenguajes de programación (incluyendo C, C++, Java, perl, tcl y python).

Postgres ofrece una potencialidad adicional sobre los sistemas de gestión de bases de datos relacionales tradicionales al incorporar los conceptos de clases, herencia, tipos y funciones que permiten a los usuarios extender fácilmente el sistema. Otras características que aportan flexibilidad al sistema son las restricciones (Constraints), los disparadores (triggers) y las reglas (rules). [19]

1.4 CONSTRUCCIÓN Y GENERACIÓN DE CONTENIDOS EDUCATIVOS PERSONALIZADOS

1.4.1 Desarrollo de contenidos en base a Objetos de Aprendizaje

El desarrollo de contenidos es uno de los aspectos más importantes en el proceso de integración de las nuevas tecnologías de la información en el aula. Dicho desarrollo incorpora una serie de pautas a tener en cuenta que no sólo implican un cambio importante en la metodología de aprendizaje sino que también se amplían las posibilidades de personalización y adaptación de los entornos virtuales de aprendizaje a las necesidades particulares y específicas de cada usuario.

El desarrollo de contenidos para el entorno de aprendizaje debe favorecer el cumplimiento de los objetivos pedagógicos establecidos por el docente para un curso. Basarse en el concepto de *Objetos de Aprendizaje* aporta importantes beneficios para el formador y el participante de una acción formativa e incorpora grandes avances y ventajas a lo largo de la fase previa de desarrollo, debido a las posibilidades de reutilización y actualización del contenido en el entorno de aprendizaje.

El diseño y desarrollo de los contenidos que se vayan a incluir en un curso de formación online requieren una adecuada estructuración y una minuciosa planificación que facilite su seguimiento por parte de los participantes en dicho curso, ya que este aspecto contribuirá a facilitar, no sólo el proceso de aprendizaje, sino también las posibilidades de control y seguimiento de dicho proceso por parte del formador. Por otro lado, solamente se puede asegurar un proceso de aprendizaje satisfactorio si dicho contenido consta además de actividades prácticas que permitan facilitar el proceso de asimilación del mismo a la vez que es posible, a través de dichas actividades, realizar un adecuado seguimiento del progreso de cada participante.

No se puede ignorar un nuevo concepto introducido a partir del desarrollo de los contenidos basados en los estándares internacionales. Se trata del concepto de “Objeto de Aprendizaje”, el cual va mas allá de lo que supone incorporar un término novedoso más al

mundo del e-learning. Desarrollar contenidos basados en objetos de aprendizaje aporta una nueva filosofía de desarrollo que, a su vez, permite garantizar la creación de programas de formación online dotados de un alto grado de personalización y flexibilidad, lo cual implica la posibilidad de cumplir objetivos de aprendizaje más específicos y al mismo tiempo adaptarse a las necesidades de cada participante.

Los objetos de aprendizaje se definen como pequeñas unidades de contenido interactivo cuya característica más importante es la posibilidad de ser fácilmente reutilizables. Dichos objetos o unidades pueden incorporar cualquier tipo de formato (impreso, Web, multimedia, Word, etc.) de acuerdo a las necesidades en cada curso. [20]

1.4.2 Consideraciones para el diseño de contenidos

En el proceso de diseño de contenidos se tiene en cuenta, como un aspecto fundamental, las características del grupo o comunidad estudiantil a la que va destinado el producto. Respecto a los usuarios, algunos aspectos importantes a considerar son: definición del grupo, edad promedio, sexo, nivel cultural, nivel de estudios, lengua, limitaciones psíquico-físicas, habilidades específicas, experiencias con sistemas e-learning, entrenamiento previo, frecuencia de utilización de herramientas telemáticas y motivación, ya que estos aspectos permiten ofrecer contenidos más personalizados, menos frustrantes y más fáciles de usar. A dichos aspectos es necesario añadir la importancia del contexto, y cuando se hace referencia al contexto se habla tanto del contexto espacial de los usuarios (y con ello su propia identidad cultural, social y económica) como al contexto que se genera en la propia red en la cual se van a interrelacionar los usuarios intercambiando información y comunicándose. La figura 3 muestra la representación de estos factores. [21]

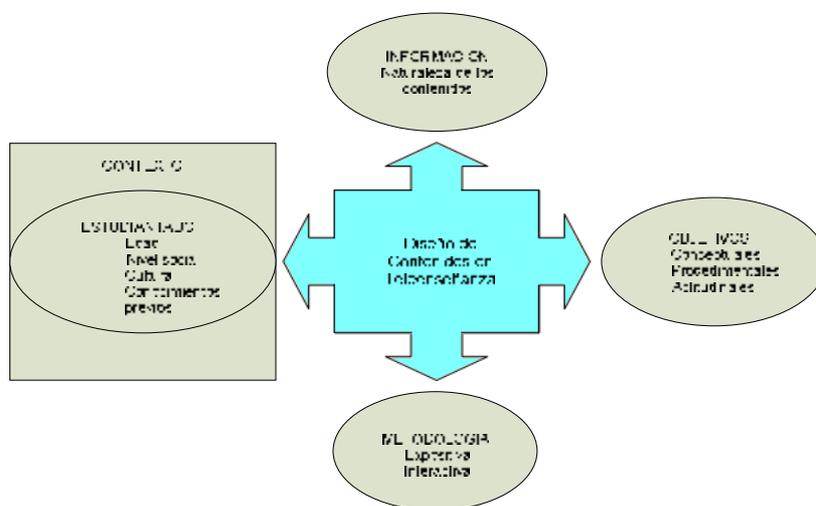


Figura 3 Factores condicionantes del diseño de contenidos en tele-enseñanza

1.4.3 Los perfiles de Usuario

1.4.3.1 El Profiling

El profiling es una técnica cuya idea principal es la de crear y gestionar perfiles. Un perfil es un modelo de un objeto (una representación compacta que describe las características más importantes del objeto), creado en la memoria del computador y es utilizado como representante (del modelo actual) del objeto en las tareas computacionales. Estas tareas incluyen por ejemplo: comparaciones, almacenamiento, resúmenes y análisis. Las aplicaciones de profiling más conocidas incluyen la personalización, la gestión de conocimiento y el análisis de datos. Para el presente trabajo la aplicación a considerar es la personalización.

1.4.3.2 El Perfil

Perfil es una palabra que viene de las palabras del latín “*pro filare*”, que significan “diseñar los contornos”. Los perfiles en el mundo de los computadores cumplen la misma tarea:

describir las características más vitales del objeto, en un formato compacto, que los computadores entiendan para que los programas los puedan usar en sus tareas. En principio, se puede hacer un perfil de todo, y por consiguiente, las características representadas en el perfil dependen de la naturaleza del objeto modelado.

En el mundo de los computadores, el objetivo de la mayor parte del profiling, es la parte más compleja e imprevisible de un ambiente informático: el usuario. Un perfil es una colección de propiedades de un objeto, en este caso del usuario. Cuando se hace un perfil de un usuario en línea, estas propiedades incluyen, por ejemplo, el método de conexión utilizado, el terminal utilizado, patrones de comportamiento e intereses del usuario. El perfil es utilizado como representante del usuario en las tareas computacionales y es utilizado en muchas situaciones, entre ellas la personalización de aplicaciones y servicios y en la automatización de tareas.

1.4.3.3 Personalización

Personalización es el conjunto de procesos en el que un sistema provee servicios basados en la información personal de cada usuario. Los tipos de personalización se dividen en muchas categorías, de las cuales cabe destacar: *reconocimiento por nombre*, *customization* y *adaptativa* [24]. El reconocimiento por nombre es la forma más básica en que el usuario es reconocido y saludado al llegar al servicio. La customization es normalmente llamada “personalización de caja de opción”, porque el usuario define los parámetros de funcionamiento del servicio, seleccionando sus preferencias de una lista de cajas de opciones. Este tipo de personalización es frecuentemente usada por los sistemas de hipermedia adaptable, en los que el perfil contiene las opciones seleccionadas.

La personalización de servicios adaptativa es la proporcionada por los sistemas de HA¹¹ (Hipermedia Adaptativa) [22]. Hacen una personalización avanzada, seleccionando el contenido de sus presentaciones, de acuerdo con las acciones del usuario anteriormente realizadas en ese servicio. Esta información es procesada y guardada en su perfil de usuario

¹¹ Los sistemas de hipermedia adaptativa (AHS) se diseñan y construyen con la intención de suministrar información hecha a la medida para usuarios individuales de acuerdo a sus preferencias.

durante la interacción del usuario con el sistema y posteriormente analizada para adaptar la presentación. Este método de personalización es el más corriente.

La personalización realizada por los sistemas de HA permite entablar una relación de marketing “uno-a-uno” real, para un gran número de usuarios. No divide los usuarios en grupos, sino que selecciona para cada usuario separadamente la presentación, utilizando para ello tecnologías de Inteligencia Artificial (IA), como son: redes neuronales, redes bayesianas, lógica difusa y razonamiento basado en casos. También se suelen emplear heurísticas¹² basadas en el dominio de aplicación para mejorar el desempeño de los sistemas. Un componente de personalización debe ser capaz de recomendar documentos (recommender systems) [23], promover productos, aconsejar apropiadamente, etc.

La personalización es cada vez más utilizada como medio de agilizar la entrega de información al usuario, convirtiendo al sistema en más útil y atractivo, de forma que el usuario se sienta estimulado al utilizarlo. Por eso, la personalización se viene convirtiendo en un requisito esencial en los sistemas de hoy en día, tales como los sitios Web de comercio electrónico y los sistemas de e-learning.

1.4.3.4 Creación de Perfiles

Hay tres métodos principales para crear perfiles: el método explícito o de creación manual; el método colaborativo o de composición a partir de otros perfiles y el método implícito, que utiliza técnicas específicas para extraer las características automáticamente. En el método explícito, los datos son introducidos por el usuario escribiéndolos directamente en su perfil de usuario, respondiendo a formularios o tests, etc. Este es el método que se va a emplear en el desarrollo del presente proyecto para la definición de los perfiles del estudiante, dejando a consideración a quien pueda interesar la implementación de los otros métodos como trabajo futuro. También se puede crear y modificar un perfil a partir de la interacción colaborativa con otros perfiles, con los que se relaciona, recurriendo a conocimiento específico del dominio y heurísticas inteligentes. En el método implícito, se extraen/crean y modifican/actualizan

¹² **Heurística:** Es un algoritmo que ofrece la posibilidad de lograr buenas soluciones en buenos tiempos de ejecución.

automáticamente los perfiles, recurriendo normalmente a técnicas de Inteligencia Artificial para realizar estas tareas. Es importante señalar que estos métodos no son cerrados y muchas veces se utilizan simultáneamente (métodos híbridos), para producir perfiles más precisos y comprensibles.

1.4.3.5 Contenido del Perfil de Usuario

El perfil de usuario contiene información modelada sobre el usuario representada implícita o explícitamente y cuya explotación permite al sistema incrementar la calidad de sus adaptaciones. Con relación al dominio un perfil se compone de las siguientes partes: una parte *independiente del dominio*, como son el nombre, edad, nivel educativo, etc. otra *dependiente del dominio*, como los intereses del usuario, sus conocimientos en una determinada área, etc. y una tercera, que no cae en la estricta definición de ninguna de las anteriores, como son las preferencias del usuario, el ambiente, etc.

Otra forma de clasificar el perfil de usuario es considerar su comportamiento con relación al cambio. Alguna información del perfil de usuario es *estática*, como el nombre, fecha de nacimiento, etc. y normalmente es introducida manualmente una sola vez por el usuario. Otra sin embargo es *dinámica*, como los intereses del usuario que cambian y por consiguiente es aconsejable que sean determinados automáticamente.

Esto significa que para obtener un perfil más actual y preciso es necesario hacer un seguimiento lo más cercano posible de las acciones del usuario, y para eso se recoge, procesa y almacena información de dichas acciones que sirva para determinar que elementos de otros perfiles está manifestando el usuario actualmente para proceder a realizar las depuraciones y actualizaciones necesarias. [24]

1.4.4 Generación de Contenidos con docbook

Para la generación de contenidos Web se presentan una serie de alternativas, cada una de las cuáles presenta ciertas ventajas frente a las demás. Sin embargo, es conveniente

homogeneizar esfuerzos y aplicar una misma metodología de creación para todos los participantes del proyecto E-LANE, dentro del cual se escogió a docbook como la mejor alternativa. [25]

DocBook es un tipo de documento XML apropiado para generar documentación técnica utilizando una estructura de libro (que puede contener capítulos, secciones, glosario, apéndices, etc.) o artículo según el tipo de documento a producir. El contenido de un documento es organizado en una estructura jerárquica de etiquetas propias de docbook. Existe un gran número de etiquetas que permiten organizar adecuadamente el contenido siendo posible incluir texto (títulos, párrafos, hipervínculos), crear listas, tablas, insertar imágenes y otro tipo de recursos como animaciones y video, además todos estos elementos pueden ser configurados correctamente gracias a que para cada etiqueta están definidos ciertos atributos a los que se les puede asignar un valor dependiendo de las necesidades del usuario. [26]

Docbook presenta una serie de características propias que hacen de él una buena elección como formato de gestión de contenidos, en comparación con otros formatos estáticos como html y procesadores de texto como Microsoft Word; estas son:

- Separación entre los contenidos y la presentación de los mismos. Lo que acarrea un gran beneficio ya que no es necesario preocuparse por la presentación sino por la información del contenido, además cuando el contenido quiera ser editado solo es necesario reconfigurar la información y no el aspecto de la misma lo cual representa un gran ahorro de tiempo.
- La Modularidad. Permite identificar las secciones con unidades didácticas y/o tareas (objetos de aprendizaje) y además ofrece la opción de almacenar cada una de las tareas en un fichero independiente del resto, de modo que el fichero principal es sólo una colección de llamadas a otros ficheros. Esto facilita la modularidad del contenido y la reusabilidad del mismo.
- Facilidad de conversión. Permite la transformación a otros formatos, tales como pdf (útil en caso de que el estudiante quiera una versión impresa del documento), rtf (formato de

texto enriquecido), texto plano, HTML (una de las transformaciones mas utilizadas), latex y poscript. Esto brinda gran versatilidad al momento de hacer uso de un mismo contenido que puede ser reaprovechado en multitud de situaciones diferentes.

- Personalización del contenido mediante hojas de estilo. Tal vez una de las bondades más grandes de docbook y en general de XML es que permite la personalización del contenido mediante hojas de estilo XSL y CSS con las que se pueden generar diferentes presentaciones de un mismo contenido adaptándolo según los requerimientos del usuario.
- Licencia libre y compatible con GPL. Este hecho no sólo aporta una ventaja de cara a la diseminación y sostenibilidad de la metodología a formular en el presente proyecto, sino que hace que la creación de contenidos esté en sintonía con el resto del proyecto E-LANE: *el uso del software libre con fines educativos*. [27]

2 ESPECIFICACIONES Y HERRAMIENTAS PARA GENERACIÓN DE CONTENIDOS

2.1 LOS ESTANDARES EN EL PROCESO DE E-LEARNING

La utilización de las tecnologías de la Información y la comunicación en la educación como un medio para difundir el aprendizaje ha hecho que múltiples instituciones educativas estén considerando el uso de algún tipo de sistema e-learning como soporte a los procesos formativos. Entre los más comunes se encuentran los Sistemas de Gestión del Aprendizaje (LMS), que están siendo desarrollados a nivel comercial por diferentes compañías que trabajan en el campo de las tecnologías de e-learning y a nivel académico por diferentes Universidades en todo el mundo.

Actualmente existe un gran número de LMS y variedad de recursos educativos lo cual representa un problema puesto que es necesaria una “metodología común” que posibilite la interoperabilidad entre estos sistemas. En algunas de las soluciones e-learning actuales generalmente los contenidos educativos preparados para un sistema no pueden ser fácilmente transferidos a otro, es decir, existe cierto grado de incompatibilidad que no posibilita la reutilización de los materiales. Esto conlleva la necesidad de establecer elementos comunes en cuanto a los contenidos educativos y a la infraestructura de los diferentes Sistemas de Gestión de Aprendizaje. [28]

Es necesario generar estándares que permitan el uso eficiente de los contenidos en cualquier plataforma. La estandarización de las tecnologías aplicadas al aprendizaje busca hacer posible la reutilización efectiva de los materiales educativos y la interoperabilidad entre los diferentes LMS; se trata de crear cursos con contenidos de alta calidad basados en el aprovechamiento de material existente y la adaptación y adecuación del entorno de aprendizaje en función de los requisitos de conocimiento y preferencias del estudiante, lo que permite aumentar el rendimiento del tiempo empleado en el proceso formativo.

Existen organizaciones que han venido trabajando en el desarrollo de recomendaciones, especificaciones y modelos que posibilitan dicha interoperabilidad; entre las más importantes se encuentran: IEEE-LTSC, ADL, IMS Global Consortium y AICC. Algunas de estas

organizaciones serán descritas brevemente en la sección 2.3, haciendo especial énfasis en ADL y su modelo SCORM.

Al hablar sobre un estándar e-learning, se hace referencia a un conjunto de reglas en común que especifican como se pueden construir cursos online y que funcionalidad deben incorporar las plataformas sobre las cuales son impartidos estos cursos de tal forma que puedan interactuar unas con otras. Estas reglas proveen modelos comunes de información para cursos de e-learning y plataformas LMS que básicamente permiten a los sistemas y a los cursos compartir datos y también brindan la posibilidad de incorporar contenidos de distintos proveedores en un mismo curso. Además, definen un modelo de empaquetamiento de contenidos estándar; los contenidos pueden ser empaquetados como objetos de aprendizaje de tal forma que se puedan crear contenidos que puedan ser fácilmente reutilizados e integrados en distintos cursos. Por último, los estándares permiten crear tecnologías de aprendizaje más sólidas que se desarrollan hasta permitir personalizar el aprendizaje basándose en las necesidades individuales de cada estudiante. [29]

2.2 ELEMENTOS A ESTANDARIZAR EN UN SISTEMA DE E-LEARNING

Los estándares e-learning se pueden clasificar en tres tipos: [30]

- *Estándares acreditados*: Son aquellos generados por organizaciones acreditadas tales como el IEEE, el CEN (Comité Europeo de Normalización) e ISO (Organización Internacional de Estándares).
- *Estándares de la industria del e-learning*: Los estándares de la industria son especificaciones y prácticas que son seguidas por casi todos los participantes en el sector. Son conocidos como 'estándares de facto'. Un ejemplo de este tipo son los producidos por el consorcio de la World Wide Web (W3C).
- *Especificaciones*: Son aquellas que describen como funcionará la tecnología. Pueden ser producidas por empresas, investigadores o consorcios. Las especificaciones que son adoptadas por la industria se convierten en Estándares de Facto.

Los diferentes estándares que se desarrollan hoy en día para la industria del e-learning se pueden clasificar en los siguientes tipos: [31]

- *Sobre el contenido o curso:* Comprende las estructuras de los contenidos, el empaquetamiento, los metadatos descriptivos y el seguimiento de los resultados.
- *Sobre el estudiante:* Comprende aspectos como el almacenamiento e intercambio de información del estudiante, competencias (habilidades) del estudiante, privacidad y seguridad.
- *Sobre la interoperabilidad:* Comprende la parte referente a la integración de componentes del LMS y la interoperabilidad entre múltiples LMS.

La figura 4 muestra una representación de la anterior clasificación:

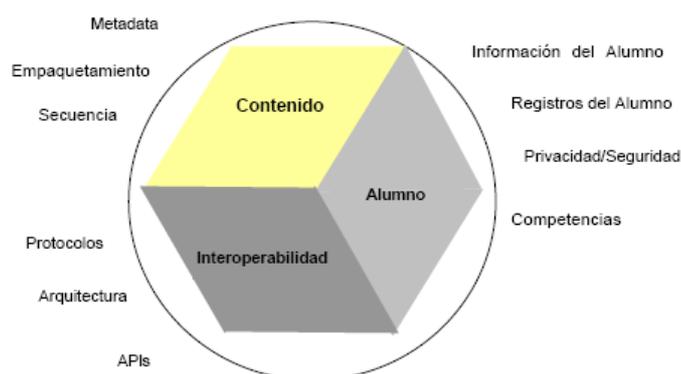


Figura 4 Clasificación de los estándares de la industria del e-learning

Mediante la aplicación de un estándar en la tecnología de e-learning se pretende lograr los siguientes aspectos:

- *Durabilidad:* El contenido debe poder utilizarse sin importar cambios en la tecnología base en la cual se elaboró. Esto sin necesidad de tener que recodificar o re-compilar programas de software.
- *Interoperabilidad:* El contenido debe ser independiente de la herramienta de creación y de la plataforma de tal manera que un mismo contenido pueda ser utilizado en diferentes plataformas.

- *Accesibilidad:* Proporcionar acceso a los contenidos desde cualquier lugar a través de un navegador en Internet sin importar la plataforma o el contenido en si mismo.
- *Reusabilidad:* Solo el uso de estándares facilitará el diseño de contenidos que puedan ser utilizados varias veces en diferentes asignaturas, cursos o programas educativos.
- *Adaptabilidad:* Los estándares se refieren al hecho de poder facilitar la adaptación o personalización del entorno de aprendizaje.
- *Productividad:* Si los proveedores de tecnología e-learning desarrollan sus productos siguiendo estándares comúnmente aceptados, la efectividad del e-learning se incrementa significativamente y el tiempo y costos son reducidos. [32]

2.3 ORGANIZACIONES IMPLICADAS EN EL PROCESO DE ESTANDARIZACIÓN

2.3.1 IEEE-LTSC

El IEEE (*Instituto de Ingenieros en Electricidad y Electrónica*) es un conocido cuerpo multinacional que desarrolla estándares internacionales para sistemas eléctricos, electrónicos, computacionales y de telecomunicaciones. El IEEE está organizado en distintos comités que se juntan y analizan las distintas tecnologías entregando como resultado una especificación o recomendación en forma de estándar. Uno de éstos es el Learning Technology Standards Committee (LTSC) o Comité para la Estandarización de las Tecnologías aplicadas al aprendizaje. [33]

El LTSC abarca todos los aspectos del aprendizaje basado en computador. Su misión principal es desarrollar estándares técnicos, prácticas recomendadas y guías para la generación de componentes software, herramientas, tecnologías y métodos de diseño que faciliten el desarrollo, implantación, mantenimiento e interoperabilidad de los sistemas educativos basados en computador.

El LTSC posee quince subcomités organizados en cinco áreas de trabajo, las cuales son: aspectos generales, aspectos relacionados con el contenido, aspectos relacionados con el estudiante, datos y metadatos y sistemas y aplicaciones de gestión. La especificación más reconocida del trabajo de IEEE-LTSC es la especificación LOM (*Learning Object Metadata*) o

Metadatos de los Objetos de Aprendizaje, que define elementos para describir los recursos del contenido. IMS y ADL utilizan los elementos y las estructuras de LOM en sus respectivas especificaciones.

2.3.2 IMS Global Learning Consortium

La misión del IMS Global Learning Consortium es brindar soporte para la adopción y el uso de tecnologías de aprendizaje a través de la Web. IMS es una organización sin ánimo de lucro que incluye más de 50 miembros y filiales contribuyentes. Estos miembros vienen de todos los sectores de la comunidad “global e-learning”. Entre estos se incluyen vendedores de Hardware y Software, instituciones educativas, publicistas, agencias gubernamentales, integradores de sistemas, proveedores de contenido multimedia y otros consorcios. [34]

IMS desarrolla y promueve la adopción de especificaciones técnicas abiertas para tecnología de e-learning. Muchas de las especificaciones IMS se han convertido en estándares de facto del Web para la generación de productos y servicios de aprendizaje. Dentro de las especificaciones IMS las más importantes son: [35]

- *IMS Learning Resources Metadata Specifications* (Especificaciones de Metadatos sobre los recursos de Aprendizaje). Genera una forma uniforme para describir los recursos de aprendizaje de manera que estos puedan ser fácilmente encontrados, por medio del uso de herramientas de búsqueda que sean capaces de interpretar estos metadatos.
- *IMS Content Packaging* (Especificación de empaquetamiento de contenido). Posibilita la creación de objetos de contenido reutilizables para que puedan ser intercambiados entre diferentes sistemas de gestión de aprendizaje gracias al empaquetamiento de dichos contenidos. Estos paquetes contienen los diferentes recursos educativos (el contenido como tal) y la información relacionada con ellos (metadatos).
- *IMS Learner Profiles Specification* (Especificación de los perfiles del estudiante). Ofrece formas de organizar la información del estudiante de manera que los sistemas de aprendizaje puedan responder de una manera más adecuada a las necesidades específicas de cada usuario.

2.3.3 ADL

En 1997 el Departamento de Defensa de los Estados Unidos y la oficina de Ciencia y Tecnología de la Casa Blanca lanzaron la iniciativa Advanced Distributed Learning (ADL). Esta iniciativa es un esfuerzo colaborativo para promover el uso de las tecnologías de la información en el aprendizaje. ADL se ha centrado desde un principio en el aprendizaje sobre la Web. Su trabajo ha acompañado al de otras instituciones, principalmente IEEE, IMS y AICC. Ha recogido “lo mejor” de las iniciativas más relevantes (el sistema de descripción de cursos en XML de IMS y el mecanismo de intercambio de información mediante un API de AICC), mejorándolas para crear su propia iniciativa: SCORM, el Modelo de referencia para objetos de contenido compartibles, que propone un entorno de ejecución, un modelo de metadatos y un modelo de la estructura de los cursos. En la sección 2.4 se realiza una descripción de este Modelo. [36]

2.4 MODELO DE REFERENCIA PARA OBJETOS DE CONTENIDO COMPARTIBLE – SCORM

SCORM es un modelo de referencia que incluye una serie de especificaciones y guías que contienen los requisitos imprescindibles que deben cumplir los contenidos educativos. Estos requisitos definen las pautas para la creación de contenidos formativos multimedia de tal manera que sea posible su reusabilidad, accesibilidad, interoperabilidad y durabilidad. SCORM reúne los esfuerzos de diferentes organizaciones en el desarrollo de un modelo general que puede ser tomado como referencia para la creación de sistemas de aprendizaje a través de Internet. [37]

La propuesta básica de SCORM es que los contenidos deben construirse a partir de los Objetos de Aprendizaje denominados SCO (objetos de contenido compartible). Para que un Objeto de Aprendizaje pueda considerarse un SCO debe ser reusable, interoperable, duradero y accesible. De acuerdo a SCORM, un SCO es un archivo digital que posee la estructura mostrada en la figura 5. La tabla 1 presenta una descripción de los componentes del SCO.

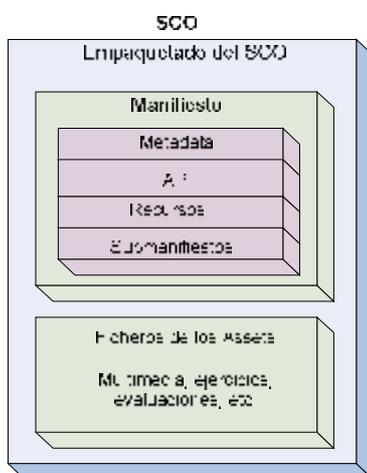


Figura 5 Estructura de un SCO [38]

SCO	
Elemento del SCO	Descripción
1. Formato del fichero	Es un fichero de extensión .ZIP.
2. Manifiesto	Es básicamente un archivo XML de nombre <i>imsmanifest.xml</i> que posee una descripción del contenido del SCO.
3. Metadata	Es información sobre el contenido, como el autor, tipo de contenido, nivel de información, copyright, etc.
4. API (Application programming interface)	Es el nombre genérico del código estándar que le permite al SCO un cierto grado de comunicación con el LMS.
5. Assets	Es el material docente o recursos que constituyen el SCO, tales como archivos de imagen, video, animaciones, etc.

Tabla 1 Descripción de los componentes de un SCO

Las especificaciones de SCORM, distribuidas por ADL, detallan cómo deben de publicarse los contenidos y usarse los metadatos; también, incluyen las especificaciones para representar la estructura de los cursos por medio de XML y el uso del API (Application Programming Interface). En general, SCORM está dividido en dos partes, el modelo de agregación de contenidos y el entorno de ejecución de SCORM.

2.4.1 Modelo de agregación de contenidos (CAM)

Contiene una guía para identificar y agregar recursos dentro de un contenido de aprendizaje estructurado. Está compuesto por los siguientes elementos: [39]

- § *Modelo de contenido:* Definición de una representación mediante XML de la estructura de los contenidos que puede utilizarse para definir todos los elementos de un curso.
- § *Metadatos:* Un mecanismo para describir aspectos específicos de los componentes del contenido. Son de dos tipos: los que incluyen la información del curso en sí, y los que contienen el material del estudiante.
- § *Empaquetamiento de contenidos:* Comprende la creación y agregación de assets (recursos), su organización y las relaciones que existan entre ellos para que puedan ser transferidos entre diferentes LMS. Todos estos datos se concentran en el archivo *imsmanifest.xml*.

2.4.2 Entorno de ejecución de SCORM (RTE)

En él se indican como se relacionan y comunican los SCO con las plataformas. Está conformado por tres aspectos: Launch, un API y un modelo de datos. [40]

- § *Launch:* Es un mecanismo que define una manera común para que los LMS puedan inicializar los recursos de aprendizaje. Este mecanismo define los procedimientos y responsabilidades para establecer la comunicación entre el LMS y el recurso o material de aprendizaje. Los protocolos de comunicación son estandarizados a través del uso de un API común.
- § *El API:* Es el mecanismo de comunicación para informar al LMS del estado de los recursos de aprendizaje (inicializado, finalizado o en condición de error) y es usado para intercambiar datos (por ejemplo, puntaje, límites de tiempo, etc.) entre el LMS y el SCO.
- § *Un Modelo de Datos:* Define el flujo de datos intercambiado entre el LMS y el contenido en el entorno de ejecución.

2.5 HERRAMIENTAS PARA LA GENERACIÓN DE CONTENIDOS

Se realiza una exploración de distintas herramientas software para la edición de contenidos con el fin de conocer el estado del arte de algunos editores y realizar un estudio sobre aquellos que soportan la DTD¹³ de docbook y por ende posibilitan la personalización y definición de rutas de aprendizaje dado que se trata de información con base en XML, la cual puede ser tratada con hojas de estilo XSL¹⁴ (Extensible StyleSheet Language). No se tiene en cuenta el soporte que tiene docBook para XML Schema [41] ya que los contenidos desarrollados en el marco del proyecto E-LANE usan la DTD de docBook y no el soporte de este tipo de documentación para XML Schema (en el momento de la elaboración de este documento, la versión reciente es DocBook W3C XML Schema V1.0b1). Además, en caso de no encontrar el editor adecuado a las necesidades planteadas por el proyecto de grado, descritas a nivel de análisis y diseño en el capítulo 3, se pretende tomar esta exploración tecnológica como una base para tener suficientes criterios en las decisiones de implementación del editor de contenidos o adaptación de un editor existente.

Se hace una exploración tecnológica alrededor de los editores con soporte para XML y en especial para docBook, al final se pretende escoger un editor basado en las comparaciones realizadas con el fin de tomarlo como referencia para el editor a integrar en el LMS dotLRN que puede ser un editor Wysiwyg¹⁵, Wysiwym¹⁶ o un híbrido entre ellos; si es el caso se utilizará solo las librerías necesarias y se implementará la funcionalidad restante, de este modo el grado de desarrollo dependerá de que tanto ofrezca la herramienta o herramientas a seleccionar en cuanto al soporte de la DTD de docBook, el lenguaje de programación en el que están desarrolladas y su arquitectura.

¹³ **DTD** (Definición de tipo de documento). Es una notación que define una estructura sintáctica válida para cierto lenguaje, estableciendo las reglas y restricciones a nivel de sintaxis para un correcto uso del mismo.

¹⁴ **XSL** está formada por dos lenguajes: **XSLT** (lenguaje de hojas extensibles de transformación), que permiten generar salidas desde bases de datos **XML**, incluir etiquetas **XHTML** y estilos **CSS** y las **XSL-FO**, que son las hojas extensibles de formato. En el 2005 ya son soportadas por algunos navegadores.

¹⁵ **WYSIWYG**: es un acrónimo para (*What You See Is What You Get*), y es usado para describir la integridad entre la apariencia del contenido editado y el producto final. Hoy es usado en procesadores de texto, en situaciones como la redacción de un hipertexto y su preparación a nivel de Web no siempre es usado [42].

¹⁶ **WYSIWYM**: (*What You See Is What You Mean*) es el paradigma creado por LyX (Un editor de texto Wysiwym). Significa que las cosas visualizadas en la pantalla de un computador deben mostrar aproximadamente la información que se está tratando de dar a conocer más que el formato actual. Pretende además reemplazar el paradigma Wysiwyg, de ahí que suene como éste último [43].

Se tiene en cuenta solo los tipos de editores Wysiwyg y Wysiwym dado que su uso es muy intuitivo y su dominio requiere poco tiempo de entrenamiento, de igual forma se considera las características de los usuarios del editor que para este caso son profesores con escaso o nulo conocimiento en lenguajes de marcado.

En [44] se realizó un estudio sobre editores para XML y se clasificaron en tres grupos diferentes: editores comerciales, de software libre o GPL (General Public License) y editores adaptables a la edición de material docente, los resultados fueron los siguientes:

- Editores comerciales: TurboXML [45], XMetal [46] y XMLSpy versión [47].
- Editores GPL: Amaya [48], HTML-KIT [49], XMLEditPro versión [50] y XMLOperator [51].
- Editores adaptables a la gestión de material docente: Course Composer versión 2.2 [52], Morphon XML-Editor versión 2.01 [53] y XML-Mind versión 0.9 [54].

En el contexto del proyecto de grado, el enfoque de esta parte de la exploración tecnológica está orientado hacia tipos de editores Wysiwyg, Wysiwym y a manera de información, editores basados en texto, así como también a editores pertenecientes al software libre y comerciales con soporte para la DTD de docBook, de esta manera el editor Amaya nombrado anteriormente debe dejarse de lado ya que a pesar de ser Wysiwyg no ofrece soporte para docbook. Por razones de uso de software libre con fines educativos (en sintonía con el proyecto E-LANE) no se dará prioridad a los editores comerciales. XML Operator tampoco está dentro de los editores Wysiwyg/Wysiwym así que también se descarta. HTML-kit, funciona sobre Windows, y se prefiere trabajar con herramientas sobre Linux ya que sobre este sistema operativo está instalada la plataforma dotLRN/OpenACS a nivel del proyecto de grado y a nivel de E-LANE en la Universidad del Cauca y es sobre Linux que mayor soporte se tiene por parte de la comunidad de desarrollo de OpenACS. La tabla 2 muestra una exploración sobre editores para la creación de contenidos en XML.

Lineamientos para Generación de Contenidos Educativos y Rutas de Aprendizaje Personalizados para el Proyecto E-LANE en la Universidad del Cauca

Editores XML Libres		
Wysiwyg	Wysiwym	Basados en Texto
VEX (http://vex.sf.net)	Conglomerate(http://conglomerate.org)	Emacs (http://www-106.ibm.com/developerwoks/xml/library/x-emacs)
	Butterfly XML (http://www.butterflyxml.org)	VIM (http://www.pinkjuice.com/howto/vimxml)
	Xerlin (http://www.xerlin.org)	Jedit (http://www.jedit.org)
	Getox for Genome(http://idx-getox.idealx.org)	Bluefish(http://bluefish.openoffice.nl/index.html)
	KXML Editor for KDE (http://kxmledito.sf.net)	
	MIView for genome (http://www.freespiders.org/projects/gmlview)	
	XML PRO by Vervet Logic (http://www.vervet.com)	
	Cooktop (http://www.xmlcooktop.com)	
	XML Buddy (http://xmlbuddy.com)	
Editores XML Comerciales		
Wysiwyg	Wysiwym	Basados en Texto
Xmetal (http://www.xmetal.org)	Stylus Studio XML Editor (http://www.stylusstudio.com)	
Editix (http://www.editix.com)	Timelux Xpress (http://www.timelux.lu)	
Authentic by Altova (http://www.altova.com)	Excsoft XML Client (http://www.excsoft.se)	
XML Editor by XMLMind (http://www.xmlmind.com)	Markup Editor 1.1.3 (http://www.topologi.com)	
i4i Tagless Editor (http://www.i4i.com)	EditML PRO by NETBRYX (http://www.netbryx.com)	
XMLWriter by Wattle Software (http://xmlwriter.net)	ElfData XML Editor (http://www.elfdata.com)	
EwebEditPRO+XML3.1 (http://www.ektron.com)	XMLware (http://www.chez.com/mustcorp/xmlware/)	
Ultra XML (http://www.webxsystems.com)	Xmloperator (http://www.xmloperator.net)	

Morphon (http://www.morphon.com)	Exchanger XML Editor (http://www.cladonia.com)	
XMLSpy (http://www.altova.com)	xmlBlueprint XML Editor (http://www.xmlBlueprint.com)	
Sintext Serna (http://www.sintext.com)	XMLBuddy PRO (http://xmlbuddy.com)	
FrameMaker (http://www.adobe.com)	EpcEdit (http://www.epcedit.com)	
Epic Editor (http://www.arbortext.com)	Xeena by IBM (http://www.alphaworks.ibm.com/tech/xeena)	
	<oXygen/>XML Editor (http://www.oxigenxml.com)	

Tabla 2 Exploración tecnológica para editores XML

Es importante tener en cuenta que los editores basados en texto no son muy adecuados para los profesores que desarrollan contenidos educativos, este tipo de editores requiere que los usuarios tengan cierto grado de conocimiento sobre XML y en especial sobre docBook lo cual implica capacitación y por ende tiempo y dinero. Por el contrario los editores tipo Wysiwyg o Wysiwym son muy intuitivos por lo que su uso se adapta a usuarios que no tengan conocimiento sobre XML, permitiéndoles concentrar sus esfuerzos exclusivamente en el contenido de los cursos que están desarrollando y no en su estructura y/o sintaxis XML.

A nivel comercial existe gran variedad de editores Wysiwyg con gran facilidad de uso, muy flexibles completos e intuitivos, tal es el caso de XMLMind. Sin embargo, el carácter de utilización de herramientas libres con fines educativos del proyecto E-LANE no va de la mano con el uso de herramientas propietarias. Por el lado del software libre, la variedad en cuanto a herramientas tipo Wysiwyg no es muy amplia, encontrando como herramienta de escritorio a VEX. También se puede incluir a quanta [55], herramienta en la cual se puede obtener vistas tanto de fuente como de presentación según se desee, pero dado que está pensada para kde y por ende diseñada con Qt¹⁷, no resulta atractiva como herramienta fácil de integrar a un framework como OpenACS.

¹⁷ **Qt:** Es un toolkit gráfico independiente de la plataforma para el desarrollo de Interfaces gráficas de Usuario.

Por otro lado existe gran variedad de herramientas Wysiwym tanto a nivel comercial como libre, a continuación se hace una breve descripción de algunas herramientas Wysiwym libres, no se describirán todas ya que existen características a nivel global que las descartan. La tabla 3 muestra dichas herramientas descartadas y las características que impiden su escogencia.

Herramienta	Característica
Getox para gnome, MIView para gnome, Conglomerate (requiere gnome 2 y las librerías de nivel respectivas), KXML por kde.	Demasiadas dependencias con respecto a una aplicación mayor como kde o gnome, además limitación forzada a usar o bien kde o bien gnome o en su defecto las librerías requeridas.
Cooktop	Herramientas únicamente para sistemas operativos Windows
XMLBuddy PRO como plugin para Eclipse.	Versiones hechas para plataformas específicas

Tabla 3 Características usadas como descarte de algunas herramientas Wysiwym

2.5.1 Butterfly

El IDE (entorno integrado de desarrollo) XML Butterfly esta construido con base en un nuevo algoritmo incremental de análisis sintáctico XML en tiempo real. El Modelo de Objetos de Documento DOM¹⁸ es actualizado en tiempo real según el usuario escriba o no en un hilo separado. Posee las características de sintaxis y resaltado de errores, la validación incremental, el autocompletado inteligente de código (basado en XML Schemas¹⁹, DTD, o análisis de documentos), XSLT pipelines²⁰, DTD y Schema Generation, y vistas simultaneas DOM y fuente. Tiene soporte para XHTML, XSL, XForms²¹, XMLSchemas, XSP²² y Cocoon sitemaps²³ al igual que soporte para adicionar fácilmente otros tipos de XML. El editor es capaz de analizar sintácticamente documentos XML que no están bien formados y mostrar la

¹⁸ **DOM (Document Object Model):** Es una forma de representar documentos estructurados (tales como una página Web HTML o un documento XML) que es independiente de cualquier lenguaje orientado a objetos.

¹⁹ **XML Schema:** es un lenguaje para describir la estructura y reglas de los documentos XML.

²⁰ **XSLT pipelines:** (procesos de transformaciones XSL conectados) facilitan las transformaciones complejas de construcción y visualización.

²¹ **XForms:** Están comprendidos de secciones separadas que describen lo que el formulario (form) hace y como luce. Esto permite opciones de presentación flexibles, incluyendo formularios XHTML clásicos, para ser agregadas a una definición de XML Form.

²² **XSP:** es la implementación de Ximian del parser de página y generador de código de ASP.NET para la ejecución de Aplicaciones ASP Web.

²³ **Cocoon sitemaps:** permite, entre otras funcionalidades, la declaración de XML pipelines. Cocoon sitemaps son una de las primeras implementaciones del concepto de XML pipeline.

fuente de los errores. Los errores en los documentos son resaltados en la fuente así como marcados en la vista DOM. Esto permite fácil conversión de HTML a XML bien formado. La vista DOM y fuente permite que la vista en árbol del XML sea actualizada instantáneamente mientras el usuario escribe. [56]

2.5.2 Xerlin

Xerlin es una aplicación para crear archivos XML. Corre sobre la máquina virtual de java 2 (JDK1.2.2 o JDK1.3). Xerlin simplifica la interacción con un archivo XML ocultando las etiquetas XML. Posee además una interfaz agradable para edición de los atributos de los elementos. Ha sido desarrollado para usuarios finales quienes no tienen la necesidad de estar familiarizados con XML, aunque también suministra una interface muy usable para quienes conozcan muy bien XML. En el momento de la elaboración de este documento la versión liberada de Xerlin era la 1.3 y se estaba desarrollando la version 2.0.

2.5.3 XML Pro v2.01

XML Pro v2.01 ofrece las características de un editor XML con una interfaz intuitiva que permite a los expertos y novatos en XML crear documentos bien formados y válidos. Está disponible sin cargo alguno para usos individuales, instituciones educativas y organizaciones sin ánimo de lucro. Es una herramienta perfecta para desarrolladores en XML, creadores de contenidos, y autores. XML Pro suministra una solución rápida y funcional de edición.

2.5.4 Borges

Borges es un sistema de producción de contenidos que tiene como meta la creación de documentos en muchos lenguajes. Cumple con metas de diseño como internacionalización, flexibilidad, contenido reusable, trabajo en equipo. Puede utilizarse para proyectos que usen documentos basados en la DTD de DocBook XML.

El sistema consta de varios paquetes aparte del paquete que tiene toda la lógica de la herramienta, el paquete Borges-doc contiene la documentación y fuentes de documentación para el sistema de gestión de documentos Borges. Los documentos están escritos con Borges, por eso las fuentes incluidas pueden usarse como un tutorial para aprender a usar la herramienta. El paquete Borges-docBook contiene el módulo DocBook para el sistema de gestión de documentos Borges, ofreciendo soporte para manipular diferentes documentos escritos con la DTD de DocBook XML. El paquete Borges-Fontend²⁴ provee una interface Web adecuada para el sistema de gestión de documentos Borges. Se trata de una interface HTML que permite a los usuarios cargar o editar en línea los diferentes módulos que tienen tareas asignadas. Los cambios se realizan en el CVS y eventualmente se pasan a las tareas asociadas.

2.5.5 Exitor

eXitor [44] es el resultado de un proyecto de investigación en el que se enmarcó un proyecto de fin de carrera, eXitor es una herramienta para la edición asistida de documentos estructurados que soporta tanto DTDs como una versión concreta de DocBook XML-Schema. Esta desarrollado en java y será liberado bajo la licencia GPL.

2.5.6 Bitflux Editor

Es un editor XML Wysiwyg soportado en el browser, completamente basado en javascript y además es de código abierto. Usa XML, DOM, XSLT, CSS y algunos esquemas XML (XML schema) pero en el momento es soportado sólo por Mozilla version >= 1.0. Las razones para haber escogido Mozilla como navegador de soporte son: Trabaja en Linux, Mac, *BSD, etc., es un browser de fuente abierta, puede renderizar XSL sobre Mac, tiene muy buen soporte para DOM/CSS, tiene herramientas muy prácticas.

Con Bitflux se puede editar contenidos semánticamente mientras se los visualiza. Se tienen todas las ventajas de XML y un procesador de palabra con presentación Wysiwyg. Implementar un editor XML soportado en un browser es ventajoso debido a que el browser es una aplicación presente en cualquier sistema operativo, cualquier persona sabe o puede

²⁴ Software que provee una interfaz adecuada para otro programa.

manejar un browser puesto que es muy sencillo y XML es el futuro en el intercambio de datos a través del Web.

2.5.7 Vex - A Visual Editor for XML

Vex es un editor para documentos XML. La palabra "Visual" viene del hecho de que Vex oculta los tags (etiquetas) XML al usuario, suministrando en su lugar una interface similar a la de un procesador de palabra (Wysiwyg). Por esto, Vex brinda un mejor soporte para documentos XHTML y DocBook.

2.5.8 Herramientas con TCL

El proyecto TclXML es una colección de herramientas y librerías para manipular documentos XML con el lenguaje de scripting Tcl. Se trata de Tcl, parsers XML basados en C, interfaces DOM y XSLT. Todos los paquetes liberados por el proyecto TclXML son de código abierto y un gran número de desarrolladores contribuyen y ayudan a mantenerlos. SourceForge suministra infraestructura y servicios de hosting (CVS, mailing lists, etc.). Hay dos aplicaciones disponibles que facilitan el procesamiento con XML: tkxmllint y tkxsltproc. Estas aplicaciones son el equivalente GUI de libxml/libxslt's xmllint y xsltproc respectivamente. Iniciando con la versión 2.3, los números de versión de los paquetes TclXML, TclDOM y TclXSLT han sido sincronizados. De ahí en adelante esos tres paquetes son liberados juntos, de ese modo debe instalarse el mismo número de versión de los paquetes. En otras palabras, si se instala paquetes diferentes entonces no hay garantía de que trabajen apropiadamente [57].

2.6 ANÁLISIS DE LAS HERRAMIENTAS ENCONTRADAS

El análisis que se realizará pretende tomar herramientas como base para la utilización, adaptación, extensión o en su defecto creación de un editor Web que soporte la DTD de docBook, idealmente se pretende escoger una herramienta y adaptarla o integrarla al LMS dotLRN, si esto no es posible, se abstraerán las características de algunas herramientas de

tal forma que se puedan utilizar en la plataforma para desarrollar un nuevo editor de contenidos que soporte la DTD de docBook.

Se establecen algunos criterios de evaluación de acuerdo a la escogencia que se pretende hacer, estos criterios se listan a continuación:

- a. *Soporte para la DTD de DocBook*: El editor debe reconocer las etiquetas de docbook y ser capaz de guardar el contenido en este formato.
- b. *Elaborado para uso a nivel de Web*: El editor debe estar elaborado en lenguajes de scripting como java script, php o similares.
- c. *Presentación Wysiwyg*: El editor debe ocultar por completo la estructura de etiquetas XML al usuario y mostrarle una vista con la presentación final del contenido.
- d. *Presentación Wysiwym*: El editor debe mostrar una vista con una presentación aproximadamente igual a la información que se está tratando de dar a conocer, la cual no corresponde fielmente al formato sobre el cual se está creando un contenido.
- e. *Adaptable para uso a nivel de Web*: El editor puede ser desplegado en un applet, es el caso de algunos editores hechos en java.
- f. *Elaborado con Tcl*: El editor está desarrollado utilizando como lenguaje de scripting a tcl.

En la tabla 4 se muestran los resultados obtenidos al someter los editores a las diferentes reglas de decisión.

	Butterfly	Xerlin	XML Pro	Borges	Exitor	Bitflux	Vex
Soporte para la DTD de DocBook	X	X	X	x	X		x
Elaborado para uso a nivel de Web				x		x	
Presentación Wysiwyg					X	x	x
Presentación Wysiwym	X	X	X			x	
Adaptable para uso a nivel de Web	X	X			X		x
Elaborado con TCL							

Tabla 4 Análisis comparativo de algunos editores XML

XML Pro puede considerarse como un editor adaptable a la edición de material docente, en este sentido es de libre uso, sin embargo no se permite la modificación del código fuente ya que se trata de un software licenciado por la empresa vervet logia. [58]

Las herramientas libres elaboradas con java son una buena opción en cuanto a soluciones soportadas en Applets y con presentación tipo Wysiwym, es el caso de Butterfly, Xerlin y Exitor. Vex por su parte ofrece una presentación Wysiwyg siendo también una herramienta adaptable a la Web mediante el uso de applets, sin embargo los applets restringen la interacción directa con OpenACS y con el sistema operativo, además el desempeño de las aplicaciones ejecutadas en applets, en comparación con las implementadas con un lenguaje optimizado para el framework usado, es menor.

Bitflux, a pesar de estar desarrollado en java script y pensado para su uso a través de la Web, no es un proyecto muy maduro en el sentido de funcionalidad y además soporta XML Schema, no la DTD de DocBook. Aunque esta propuesta de usar XML Schema es mucho más ambiciosa que el uso de DTD, para el caso del proyecto E-LANE en la Universidad del Cauca se necesita herramientas que funcionen acorde a las tecnologías usadas actualmente al interior del proyecto, en este caso la DTD de docbook. No por ello se deja de lado el uso futuro de XML Schema por ser más orientado a XML y estar llamado a reemplazar a la validación mediante DTDs, aparte del soporte para XML Schema que está generando la comunidad de desarrollo de DocBook [59].

Además del análisis realizado sobre los editores anteriores, también se tuvo en cuenta los desarrollos a nivel del framework OpenACS, como los paquetes Wimpy Point, xowiki, edit this page, wiki, entre otros. Todos estos paquetes permiten la generación y edición de documentos con presentación en html, destacándose Wimpy Point y xowiki. Xowiki es una implementación wiki para OpenACS implementada en xotcl, esta herramienta es excelente, permite la creación y edición de varios tipos de páginas de acuerdo a la imaginación del diseñador, la creación de directorios adecuados para la organización de las páginas del sitio wiki, entre otras muchas ventajas [60].

Wimpy Point es un paquete que permite la creación de presentaciones en las que el contenido introducido se almacena en el repositorio de contenidos de OpenACS y es

presentado como Slides. Wimpy Point es una aplicación Web que constituye un editor de contenidos Wysiwym y ofrece al usuario una previsualización del contenido que está creando. Es una herramienta con interfaces de usuario muy intuitivas en las que los hipervínculos constituyen la forma de navegación por la aplicación. Este editor permite la inserción de texto, hipertexto y recursos como imágenes, animaciones, videos y cualquier archivo (por ejemplo, .doc, .pdf, etc.) con contenido que necesite ser publicado como parte de la presentación creada.

Finalmente, después del proceso de evaluación de los editores para la generación de contenidos descritos en este documento, la solución adoptada por el equipo de trabajo consistió en utilizar como editor base la aplicación implementada por el paquete Wimpy Point de OpenACS debido a su estructura de almacenamiento de contenidos haciendo uso del repositorio de contenidos (content-repository) de OpenACS. En el momento de liberarse una versión estable de xowiki, ya se tenía gran parte del editor implementada, así que se decidió continuar con el trabajo, claro está que sacando provecho de algunas características de xowiki con el uso de AJAX y xotcl para el desarrollo de los subsistemas restantes además del editor.

El paso siguiente fue diseñar el editor con soporte para la producción de contenidos educativos con base en objetos de aprendizaje siguiendo la estructura de documentación planteada por DocBook y con base en el repositorio de contenidos, al igual que Wimpy Point. Dicha estructura sugiere la segmentación de los contenidos con el fin de aumentar la reusabilidad de los mismos, es decir, que se maneje una estructura jerárquica en forma de árbol en la que los contenidos constituyen un libro, dividido en capítulos, cada uno de los cuales está constituido por secciones. La idea del concepto de segmentación es que cada tema del contenido constituya una sección y físicamente corresponda a un archivo con estructura docbook. Por lo tanto, la información y los recursos (archivos adicionales que se agregan al contenido, como imágenes, animaciones, video, etc.) van a estar contenidos en las secciones y cada capítulo será un índice de las secciones que a él pertenecen y el book (que es un único archivo por cada curso) constituirá un índice de los capítulos en los que haya sido dividido el curso.

Muchas de las funcionalidades de Wimpy Point han sido incluidas en RACE Editor²⁵ realizando cambios en la lógica del modelo de datos del paquete original para adecuarlo a la nueva estructura de contenidos utilizada como también en la lógica de control y presentación. De ese modo hubo necesidad de una reescritura completa de Wimpy Point. Adicionalmente fue necesario definir interfaces que permitan implementar (por parte del subsistema RACE-Agent descrito en el capítulo 4) toda la lógica requerida para ensamblar los contenidos adecuadamente, procesarlos y empaquetarlos de acuerdo a SCORM para finalmente publicarlos en el sistema de repositorio de contenidos de aprendizaje (LORS) de dotLRN. RACE Editor es una herramienta enfocada a ser utilizada por los profesores para producir contenidos en la plataforma en base a objetos de aprendizaje y constituye un módulo o subsistema de todo lo que se realizó en este trabajo de grado. Los nombres de los subsistemas del sistema RACE se escogieron en inglés debido a que este es el idioma base en el cual están contruidos todos los paquetes del Framework OpenACS

En parte, el presente proyecto de grado se planteó como una alternativa al proceso en que actualmente se están desarrollando y adecuando los contenidos para que puedan ser publicados en el entorno virtual de aprendizaje (EVA). Dicho proceso se basa en el uso de varias herramientas libres a nivel de escritorio, estas herramientas son: XMLMind [54] para la generación de contenidos en formato docbook, el procesador xsltproc para la transformación de los documentos DocBook a HTML mediante hojas de estilo XSLT para que puedan ser vistos adecuadamente en el navegador del cliente y Reload Editor [61] para empaquetar el contenido de acuerdo a SCORM. Para el manejo de estas herramientas se requiere de personal capacitado. La idea es desarrollar la lógica necesaria para que el profesor pueda ejecutar desde su interfaz de usuario un proceso que implementa la funcionalidad desarrollada por las herramientas anteriores. De esta forma, el proceso completo desde la creación hasta la publicación del contenido lo realiza el profesor directamente en el entorno virtual sin necesidad de utilizar herramientas de escritorio y de forma transparente, es decir, sin necesidad de tener conocimientos sobre las tecnologías (XML, XSLT, SCORM, etc.) involucradas en el proceso.

²⁵ **RACE:** Rutas de Aprendizaje sobre Contenidos Educativos

En cuanto al empaquetamiento de los contenidos de acuerdo a las especificaciones de SCORM, también existe una implementación de Reload que funciona a nivel Web y resulta muy interesante, es similar en cuanto a soporte para el empaquetado y recibe el nombre de WeLoad [62].

El siguiente capítulo contiene las pautas que deben ser consideradas por el profesor al momento de diseñar contenidos con RACE Editor, en especial objetos y actividades de aprendizaje teniendo en cuenta el perfil del estudiante.

3 GENERACIÓN DE CONTENIDOS Y RUTAS DE APRENDIZAJE PERSONALIZADOS

3.1 ELEMENTOS CLAVES DEL PERFIL DE UN ESTUDIANTE EN UN ENTORNO DE APRENDIZAJE MEDIADO POR TECNOLOGÍAS TELEMÁTICAS

A continuación se presenta la descripción del proceso de caracterización de los elementos claves del perfil de los estudiantes dentro del proyecto E-LANE en la Universidad del Cauca. Los elementos identificados son el estilo de aprendizaje, la tecnología de acceso al entorno virtual de aprendizaje y la Usabilidad de los contenidos.

3.1.1 Estilo de aprendizaje

La caracterización del estilo de aprendizaje se basa en el concepto de la psicología cognitiva²⁶ de estilos de aprendizaje, aspecto inherente a todo estudiante de cualquier área del conocimiento. El término “estilo de aprendizaje” se refiere al hecho que cada persona utiliza su propio método o estrategia al momento de asimilar la información. Aunque las

²⁶ **Psicología Cognitiva:** Es el ámbito de la psicología que estudia el pensamiento, el proceso mental que está hipotéticamente detrás del comportamiento. Este cubre una amplia gama de temas de investigación, examinando preguntas sobre los trabajos de la memoria, percepción, atención, razonamiento y otros procesos cognitivos superiores.

estrategias varían según lo que se quiera aprender, cada uno tiende a desarrollar ciertas preferencias o tendencias globales, tendencias que definen un estilo de aprendizaje.

No existe una única definición del término estilos de aprendizaje sino que son muchos los autores que dan su propia definición, como por ejemplo la que se presenta a continuación formulada por el Investigador David Kolb: "*Los estilos de aprendizaje son los rasgos cognitivos, afectivos y fisiológicos que sirven como indicadores relativamente estables, de cómo los estudiantes perciben, interaccionan y responden a sus ambientes de aprendizaje*" [63]. Los rasgos cognitivos tienen que ver con la forma en que los estudiantes estructuran los contenidos, forman y utilizan conceptos, interpretan la información, resuelven los problemas y seleccionan medios de representación (visual, auditivo y kinestésico). Los rasgos afectivos se vinculan con las motivaciones y expectativas que influyen en el aprendizaje, mientras que los rasgos fisiológicos están relacionados con el biotipo²⁷ y el biorritmo²⁸ del estudiante [64]. El estilo de aprendizaje se entiende como la forma en que el estudiante percibe y procesa la información y se centra en las fortalezas y no en las debilidades del mismo. Cabe aclarar que no existe estilo de aprendizaje correcto o incorrecto sino que todos tienen validez desde la individualidad del aprendiz.

En los últimos años se han desarrollado numerosas investigaciones en este campo y para llevar a la práctica la teoría de los estilos de aprendizaje los autores han elaborado instrumentos y herramientas que posibilitan este diagnóstico. Entre algunos de los estudios realizados se pueden citar:

- a) El modelo de Felder - Silverman. "Index of Learning Styles (ILS)".
- b) El modelo de David Kolb.
- c) El modelo de Rita y Kenneth Dunn.
- d) El cuestionario de estilos de aprendizaje CHAEA de Honey-Alonso.

Si bien es cierto que cada autor describe los estilos de aprendizaje de forma diferente, existen puntos convergentes y similitudes entre las características del estudiante de acuerdo

²⁷ **Biotipo:** Grupo natural de individuos con la misma composición genética, o patrimonio hereditario homogéneo. Equivalente a raza fisiológica.

²⁸ **Biorritmo:** Ciclo periódico de fenómenos fisiológicos que en las personas puede traducirse en sentimientos, actitudes o estados de ánimo repetidos cada cierto tiempo.

Lineamientos para Generación de Contenidos Educativos y Rutas de Aprendizaje Personalizados para el Proyecto E-LANE en la Universidad del Cauca

con los estilos de aprendizaje de preferencia. Además, es sabido que ningún estudiante posee un único estilo de aprendizaje sino que combina elementos de al menos dos de los estilos definidos por el modelo tomado como referencia y que el aprendizaje no solo depende del contenido que se aprende y la individualidad psicológica y física del aprendiz sino también del medio ambiente que lo rodea (condiciones de sonido, luz, temperatura, etc.). Otros autores incluso sugieren el término '*preferencias de estilos de aprendizaje*' que es más concreto y hace referencia a cómo el educando se siente más cómodo al estudiar y aprender: solo o en grupo, con o sin música, escuchando o leyendo, etc., más allá de si el estilo es o no eficaz.

Una vez realizado un análisis de cada uno de los modelos anteriormente mencionados, básicamente de los tests generados por cada autor, se hizo un proceso de selección para definir el modelo a utilizar en el desarrollo de este proyecto. Los criterios tenidos en cuenta fueron los siguientes:

Modelo Criterio	<i>Modelo Felder Silverman</i>	<i>Modelo de Kolb</i>	<i>Modelo de Rita - Kenneth Duna</i>	<i>CHAEA Honey - Alonso</i>
<i>Fecha realización</i>	1988	1984	1985	1994
<i>Popularidad</i>	Alta	Alta	Muy Alta	Alta
<i>Extensión del test (número de preguntas)</i>	44	12	4 factores – 18 elementos	80
<i>Objetividad en la formulación del test</i>	Media	Alta	Alta	Media
<i>Consideración del entorno del estudiante</i>	No	No	Si	No
<i>Alternativas al test</i>	No	Si	Si	No

Tabla 5 Comparación de los Modelos de Estilos de Aprendizaje

Como podemos ver en la tabla anterior, el modelo de Rita-Kenneth Dunn es el único que considera aspectos del entorno o ambiente del estudiante lo cual representa una gran ventaja sobre los otros modelos mencionados. En cuanto al cuestionario utilizado en el modelo CHAEA, es muy asequible y a pesar de que tiene muchas preguntas no muy específicas, pues dependen del contexto de quien resuelve el test, permite aproximarse a estilos de aprendizaje bien definidos. El test de Felder [65] tiene preguntas más independientes del contexto y presenta ciertas analogías con otros tres modelos de estilos de aprendizaje: el modelo de Kolb y otros dos modelos, uno basado en el indicador de Myers-Briggs y el otro una aplicación del instrumento de Herrmann basado en la especialización de los hemisferios del cerebro. Kolb es un test relativamente fácil de entender por cualquier estudiante y el cuestionario que propone está compuesto de preguntas simples y en número menor que el resto de tests considerados [66].

Se utilizará a Kolb como test de referencia en el prototipo a implementar y se complementará con recomendaciones dadas al estudiante a partir de los estudios de las hermanas Dunn, también, se suministrará un pequeño cuadro explicativo con una imagen que muestre las características principales de cada perfil, de ese modo el estudiante podrá ubicarse manualmente en el estilo con el cual se sienta más identificado. Los estudiantes que pueden situarse dentro de un estilo específico son muy escasos, sin embargo pueden existir, para este tipo de estudiantes puede ser muy útil comprobar si su estilo es el adecuado mediante la información breve que describe dicho estilo (información que será igualmente suministrada).

La siguiente tabla muestra los estilos de aprendizaje definidos por cada modelo:

	Modelo Felder - Silverman	Modelo de David Kolb	Modelo Dunn y Dunn		CHAEA Honey – Alonso
Estilos propuestos	Sensitivos – Intuitivos	Convergente	<i>Estímulos</i>	<i>Elementos</i>	Activos, Reflexivos, Teóricos y Pragmáticos
	Visuales – Verbales	Divergente Asimilador Acomodador	Ambientales	sonido, luz, temperatura, diseño.	

	Inductivos – Deductivos		Emocionales	motivación, persistencia, responsabilidad, estructura
	Secuenciales – Globales		Sociológicos	solo, en grupo
	Activos - Reflexivos		Físicos	percepción, alimento, tiempo, movimiento
			Cognitivos	global, analítico, impulsivo, reflexivo

Tabla 6 Clasificación de los estilos de aprendizaje de cada modelo

En el anexo B se encuentra una descripción detallada de cada uno de los Modelos y sus respectivos Estilos de Aprendizaje.

3.1.2 Tecnología de acceso del estudiante

Para identificar la tecnología que posee el estudiante se tendrá en cuenta el tipo de navegador, el sistema operativo que esté usando y la velocidad de conexión (conectividad). El tipo de navegador es importante debido a que existen algunas etiquetas que no son mostradas en Internet Explorer mientras que en Firefox sí, este aspecto es fácil de comprobar mediante los validadores²⁹ de la W3C [67], esto permite que al encontrar alguna incompatibilidad en algún tipo de navegador, pueda ser corregida implícitamente de acuerdo al navegador identificado sin que el estudiante tenga que hacer nada adicional al acceder a un determinado contenido.

Conocer el sistema operativo resulta importante, ya que esto ayuda a determinar que tecnologías pueden ser soportadas, por ejemplo, no existe Internet Explorer para Linux. En cuanto a la velocidad de conexión, es indispensable conocerla ya que un ancho de banda bajo no será apto para el envío de video o audio (wav), solamente imágenes y texto.

²⁹ Existen validadores como: <http://webxact.watchfire.com/>, <http://validator.w3.org/checklink>, <http://www.htmlhelp.com/tools/validator/> y <http://www.doctor-html.com/RxHTML/>, entre otros.

3.1.2.1 Identificación del navegador y el sistema operativo



Figura 6 Estimación del uso de navegadores Web con respecto al tiempo

Existen muchos navegadores Web [68], sin embargo se hará especial énfasis en los más utilizados como se muestra en la figura 6, relacionada con la guerra de los navegadores³⁰, que son Internet Explorer y Mozilla, junto con los basados en este último como Firefox. La gran variedad de navegadores presentan ciertas compatibilidades así que pueden identificarse ya sea con Internet Explorer ó con Mozilla (o Netscape).

Para la identificación del tipo de navegador y/o sistema operativo existen varias técnicas:

- Utilizar los métodos del API para TCL de AOLserver, por ejemplo ns_conn.
- Utilizar JavaScript.
- Utilizar los logs de OACS.
- Utilizar un híbrido de los métodos anteriores.

Al identificar el sistema operativo se podrían sacar estadísticas útiles acerca de los visitantes de un sitio Web en general, igualmente conocer el tipo de navegador permite enviar la

³⁰ El término "Browser War" (Guerra de los navegadores) es el nombre dado a la competencia por dominar el mercado de los navegadores Web. El término es más comúnmente usado para referirse a dos períodos específicos de tiempo: La intensa lucha particular entre Internet Explorer y Netscape Navigator durante la década de los 90s, y la creciente amenaza que representa Mozilla Firefox a Internet Explorer desde 2004 hasta 2005.

página apropiada para que se visualice de forma correcta, esto dado que cada navegador tiene su propia forma de mostrar html o ejecutar código Javascript. En este trabajo de grado no se realiza optimización de páginas Web de acuerdo al navegador ni se llevan estadísticas de visitantes, ya que estos aspectos no resultan primordiales para el desarrollo exitoso de los objetivos planteados.

3.1.2.2 Identificación de la velocidad de conexión

La identificación de la velocidad de conexión desde donde el estudiante accede a dotLRN resulta un dato valioso al momento de presentar los contenidos a dicho estudiante, esto permite determinar qué tipo de recursos de cada contenido es soportado, de acuerdo a su tamaño, por la velocidad de conexión que tenga disponible el estudiante. Así por ejemplo con una conexión por módem a 56 kbps, el envío de un video (mpeg, avi, wav, etc.) requerirá mucho tiempo para descargarse ó bien para verse directamente en el navegador mediante plugins de programas como Real Player y Windows Media Player, en este caso este recurso se “ocultará” al estudiante y se presentará recursos de tamaño pequeño como texto o imágenes con información adicional sobre lo que se pretendía transmitir con el recurso descartado o mejor aún se puede presentar una versión del video con una calidad menor optimizada para esta velocidad.

La identificación de la velocidad de conexión puede hacerse explícitamente mediante una ventana emergente o un formulario en el cual el estudiante ingresa los datos del tipo de conexión que posee y la velocidad de la misma, sin embargo lo ideal es detectar la velocidad de conexión de forma implícita. Para esto se podría utilizar medidores de velocidad en línea [69], medidores de velocidad de conexión ofrecidos a manera de servicio Web gratuito ó implementar un medidor de velocidad propio.

Una manera de establecer la velocidad de conexión de forma automática es descargando un archivo, por ejemplo una imagen, y determinando cuando la transferencia se ha completado, conociendo el tamaño de la imagen y el tiempo que tarda en descargarse se puede saber la velocidad de conexión de forma automática [70]. Sin embargo, este método no es muy exacto ya que en realidad la descarga de una imagen de tamaño moderado sólo utiliza el

100% del ancho de banda de descarga después de un cierto tiempo inicial como se ve en la figura 7:

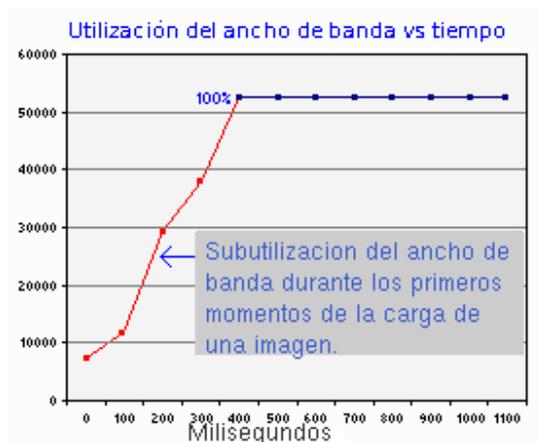


Figura 7 Utilización del ancho de banda al inicio de una descarga

Otro aspecto importante a tener en cuenta son los lenguajes utilizados para realizar los cálculos, así, si se usa JavaScript, el objeto `date()` de JavaScript suministra unidades de milisegundos, pero no necesariamente milisegundos exactos, por ejemplo sobre Windows la exactitud es apenas 0.01 segundos (en el mejor de los casos). Esto significa que dependiendo del tiempo final de descarga hay únicamente dos o tres dígitos de precisión [71]. Las siguientes definiciones son muy usadas en las pruebas de velocidad y ancho de banda de una conexión:

- **QoS:** Representa la capacidad de descarga coherente suministrada de acuerdo al ancho de banda. Si el porcentaje de QoS es alto, la conexión a Internet también será alta.
- **RTT:** El Round Trip Time (tiempo de ida y regreso) muestra el tiempo total en milisegundos (ms) para enviar un pequeño paquete de datos durante la prueba de velocidad de la conexión y obtener una replica de regreso. Si el RTT es rápido (pequeño) la velocidad de la conexión será mayor.
- **Max Pause:** Max Pause es la mayor pausa grabada durante la descarga de datos en la prueba de velocidad de la conexión. Esta debe ser un número muy pequeño, sino, puede indicar congestión o fallas en la velocidad de la conexión.

En general, existe cierto número de técnicas para medir la velocidad y el ancho de banda, la mayoría de ellas pueden clasificarse dentro de dos grupos: un grupo se basa en el envío de un paquete, la desventaja de este modelo es la gran sobrecarga en el consumo del ancho de banda. Otro grupo se basa en el envío de un par de paquetes, se presenta baja sobrecarga en comparación con el primer grupo [72].

Se empleará una aplicación ya construida de libre uso desarrollada con java que no tiene los problemas de precisión presentados por JavaScript. También se tendrá en cuenta que la técnica usada se relacione con el envío de paquetes y no con la descarga de imágenes para mayor exactitud.

De igual forma sería de gran utilidad determinar la dirección IP desde la cual el estudiante está accediendo a dotLRN, su importancia radica en poder tener un criterio primario de selección del tipo de contenido a presentar al estudiante, es decir, si se sabe que el estudiante se está conectando desde Intranet, entonces se le presenta todos los recursos de un contenido determinado; sino, se pasa a evaluar otros criterios a partir de la identificación de la velocidad de conexión del estudiante, mediante los cuales se puede restringir o no el acceso a recursos de gran tamaño como videos de alta calidad. Para determinar la IP se puede usar aplicaciones o servicios propios de OpenACS como la aplicación 'Monitor' `/acs-admin/monitor`, que determina la actividad del servidor y la IP desde donde ha sido accedido. También pueden usarse métodos de aolserver (`ns_conn`), los logs de OpenACS o bien procedimientos tcl definidos dentro de la aplicación, JavaScript, entre otros [70].

3.1.3 Usabilidad de los contenidos

- *Empleo del color en el diseño Web*

En el momento de utilizar color en el diseño Web hay que tener en cuenta su utilidad, su significado, su capacidad para comunicar y transmitir sensaciones y el entorno cultural, pero también las convenciones uniformemente aceptadas en el diseño de interfaces de aplicaciones informáticas. Por otra parte, el efecto emocional del color difiere según el

entorno cultural. Como ejemplo de opuestos: para simbolizar el luto por la muerte de una persona, en occidente se usa el color negro mientras que en la cultura japonesa para el mismo motivo se utiliza el blanco.

Una división de los colores diferencia colores cálidos como naranja, rojo, amarillo que transmiten sensación de intimidad, placer, cercanía, de los colores fríos como azul, violeta o verde, que producen una sensación de distancia, y frialdad. (Se llega a diferenciar un tipo frente a otro por su efecto sobre la presión sanguínea de la persona que percibe el color: los cálidos la aceleran, mientras los fríos la relajan) [73].

Los colores más utilizados en las aplicaciones informáticas son los grises y azules. Tonos fríos que transmiten relajación y más adecuados cuando vamos a pasar largas horas ante un monitor. Sin embargo, en el caso de las personas con discapacidades visuales relacionadas con la percepción de color se deben utilizar recursos adicionales para la comprensión. En contexto de interfaz de aplicaciones son convenciones en uso de color:

Gris: Estado de inactividad, deshabilitado.

Verde: Correcto, permitido, no crítico.

Amarillo: Advertencia, cuestión crítica, alerta.

Rojo: Error, parada, prohibición.

Azul: En interfaz de navegador corresponde con el color por defecto de los hipervínculos de una página Web.

Un buen criterio es usar colores poco saturados³¹, en general suaves y solamente en pequeñas áreas colores impactantes. En general, el significado de los colores puede tomarse como sigue:

Rojo: es un color que destaca en cualquier composición. Es un tanto agresivo ejemplo: Ferrari Rojo. Transmite: excitación, pérdidas financieras, peligro, alto, error, sexualidad, pasión, sangre. Activa la circulación, eleva la presión arterial y acelera la respiración.

Amarillo: El amarillo es el color del sol, de la luz. Transmite sensaciones alegres pero también sirve para transmitir avisos, precaución. (Muy utilizado en las señales de tráfico).

³¹ El color tiene tres atributos cuyo juego permite crear variaciones: Tono: calidad del color; luminosidad: grado de blancura o reflectancia a la luz, saturación: pureza del color, grado de fuerza/carga de un color concreto.

Verde: el color de la naturaleza, la vegetación. Transmite frescor, salud, los verdes azules son los favoritos de los consumidores. Simboliza "un paso más": correcto, avanzar (semáforos). Es un color sedante, ayuda al reposo y la vista.

Azul: el color de la credibilidad (en la mayor parte de los informativos el tono predominante de fondo es el azul, así como en comunicaciones corporativas o gubernamentales): transmite seriedad, credibilidad, serenidad, tranquilidad. En su justa medida, potencia la paciencia y la amabilidad.

Negro: color del luto en los países occidentales: Misterio, lujo, poder, elegancia.

Púrpura: el color de la riqueza, del poder. (Emperadores Romanos, Cardenales de la Iglesia, Reyes): riqueza, lujo, inteligencia, sofisticación, misterio.

Blanco: luto en los países orientales. Transmite pureza, limpieza, luminosidad pero también vacuidad (vaciedad-falta de contenido y privacidad). Es muy utilizado en diseños minimalistas.

Gris: Es el color más neutro e incluso aburrido. Expresa elegancia, aburrimiento, vejez.

En cuanto a los errores a evitar en el empleo del color en monitores, al diseñar para pantalla de computador, conviene evitar la utilización de colores demasiado saturados, utilizar una paleta de colores extensa, poco contraste de contenidos y elementos de lectura, esto dado que existe una tendencia de diseño de presentar en contenidos, los textos en tonos grisáceos sobre fondo blanco, lo que genera algunos problemas de legibilidad. Ejemplo incorrecto, texto verde sobre fondo amarillo; ejemplo correcto: texto negro sobre fondo claro [74]. Es más, se recomienda utilizar lo mínimo posible los colores de fondo en diseño Web, el fondo blanco es mucho más efectivo [75].

A nivel del desarrollo de contenidos para E-LANE en la Universidad del Cauca, se recomienda tener en cuenta las características a nivel general para el empleo del color, un estudio más profundo del uso de los colores en el diseño Web y en el desarrollo de contenidos está fuera del alcance de este trabajo de grado ya que existen personas especializadas en el tema como son los diseñadores gráficos. De ese modo se utilizarán las hojas de estilo en cascada definidas para E-LANE [76] por los expertos en el tema, sin embargo se dejará la opción de definir y aplicar hojas de estilo en cascada propias y por ende colores personalizados.

3.2 ACCESIBILIDAD Y USABILIDAD EN EL DISEÑO DE CONTENIDOS PERSONALIZADOS

Accesibilidad y Usabilidad son dos de los conceptos más importantes para los desarrolladores Web. Accesibilidad significa que los sitios Web están libres de barreras para todos y abiertos para personas con alguna discapacidad. Usabilidad hace relación a la facilidad de uso de un sitio Web. Tanto la accesibilidad como la usabilidad son poderosos indicadores cuando se trata de diseñar sitios Web de manera universal, estos indicadores reflejan el grado de satisfacción de los usuarios al tiempo que contribuyen a su incremento [77].

Un sitio accesible permite que todos los usuarios lo recorran, sin tener en cuenta tipos de navegadores, resoluciones, configuraciones o la vista. Los sitios accesibles están abiertos para los 750 millones de discapacitados del mundo, incluyendo a la numerosa población ciega. Mediante una mejor autoría, los proveedores de información de la Web, pueden evitar el cierre de la era de la información a los discapacitados del mundo y, al mismo tiempo, hacerse de una buena porción de este mercado en gran parte olvidado [78].

Los sitios Web accesibles ofrecen mayor flexibilidad para el crecimiento futuro de la Web. Aunque la Web está considerada como un medio visual, las páginas accesibles se prestan para permitir presentaciones audibles y en Braille. Hoy en día son comunes los navegadores sonoros entre las personas con limitación visual y pueden volverse muy populares en el futuro a través de navegadores Web móviles, navegadores telefónicos o del "Web Walkman". Si los autores escriben páginas accesibles darán a sus clientes y usuarios la flexibilidad necesaria para adaptarse a estas tecnologías innovadoras [79].

En el marco del proyecto de grado, se tendrá en cuenta la accesibilidad aunque no de manera total sino como aspecto de mejoramiento de las interfaces de los estudiantes específicos para quienes va dirigido el sistema, de este modo se considerará incluir un atributo "ALT" ó de texto alternativo suficientemente rico de tal manera que reemplace efectivamente la información de la imagen, sonido ó video de gran tamaño, en caso de no tener un reemplazo del mismo tipo con calidad menor pero más liviano. Además esto habilita

una adecuada información en caso de estudiantes que debido a un reducido ancho de banda y velocidad deban navegar usando browsers tipo texto como es el caso de 'links', por citar uno.

A nivel de los desarrolladores se tendrá en cuenta que los contenidos generados con el editor sean validados con validadores como [80] ó los de la W3C, tanto para HTML como para CSS. Adicionalmente se pueden ver los contenidos generados en diferentes resoluciones, intensidades de colores, tamaños de fuentes o ventanas, esto junto con la validación da independencia de la plataforma. Las páginas de los contenidos bien elaboradas se ajustarán y permanecerán accesibles para cualquier ambiente de navegadores. Durante el diseño del editor se cuidará que el html que genere, en caso de querer generarlo a partir de la fuente en docBook, tenga solamente la estructura del contenido, la presentación será dada mediante hojas de estilo en cascada.

También se puede comparar las páginas Web en diferentes navegadores (sin dejar de incluir algún navegador de sólo texto como Lynx, links, etc. o configurados para trabajar en modo solo texto), además si se quisiera llegar a una mayor cantidad de personas se debería pedir a alguien que lea en voz alta los contenidos, para simular lo que escucharía una persona con limitación visual o quien tenga un sistema de navegación en movimiento, sin embargo este será un elemento adicional que no se tendrá en cuenta, no por carecer de importancia sino por que se conocen los perfiles hacia los que van a ir dirigidos los contenidos, de todos modos en un sistema accesible son cosas que no pueden obviarse. Además existen también otras consideraciones que no se toman como prioritarias en este trabajo de grado pero que es importante conocerlas, es el caso de personas que acceden a la Web por diferentes medios, donde deben evitarse frases que crean dependencia de la plataforma, por ejemplo "haz clic aquí" es inapropiada para alguien sin un ratón, también es inútil cuando el ancla de un enlace se lee en voz alta o se imprime al final del documento como resumen, otro ejemplo son frases como "ver abajo" ya que "abajo" carece de sentido cuando un documento se lee en voz alta.

Dos herramientas de utilidad para quien desarrolla contenido Web, durante la evaluación de la independencia de plataforma de un sitio, son 'Web Page Purifier' [81] [82] y 'Web Page

Backward Compatibility Viewer' [83] (*Purificador de Páginas Web y Visualizador de Compatibilidad de Páginas Web*).

En cuanto a la usabilidad, el código html que docBook genera, garantiza un diseño centrado en el usuario, en especial la presentación dada por la Universidad Carlos III de Madrid, en la cual se abarcan aspectos como la navegación entre las diferentes páginas (navegabilidad), la facilidad de encontrar un elemento dado dentro del sitio (relacionada con la arquitectura de la información), el número de errores del sistema (relacionado con aspectos de programación), la legibilidad de los textos (correspondiente al diseño Web), la correcta transmisión de la información, además del acceso a dicha información por todos los posibles usuarios (accesibilidad). Para validar la usabilidad de los contenidos generados a partir de docBook la mejor forma de medirla es observar el uso de la aplicación Web por los usuarios, anotando sus reacciones ante la interfaz, las tareas más comunes que realizan, los errores cometidos en ellas y el grado de satisfacción obtenido en su contacto con los diferentes contenidos [84].

3.3 SISTEMAS RECOMEDADORES COLABORATIVOS PARA MATERIAL DE APRENDIZAJE MULTIMEDIA

Los sistemas recomendadores (encomender systems) son usados para ayudar a los usuarios a encontrar la información que consideran importante en un ambiente en donde la cantidad de información en el mundo se incrementa. Los encomender systems pueden basarse en el análisis de contenidos o en el filtrado colaborativo. En el primer caso, el contenido de un documento es analizado automáticamente para extraer las características relevantes mediante la aplicación de diferentes medidas, y las características se comparan con las que se desean obtener. En el segundo caso los sistemas basan sus decisiones en opiniones dadas por los usuarios de manera previa. Ambos tipos de sistemas usan metodologías de modelamiento centradas en el usuario. En un ambiente de aprendizaje soportado en computador que consiste de muchos servidores localizados en diferentes partes del mundo y que además contienen material de aprendizaje que cambia muy rápidamente, un usuario tendrá algunos problemas localizando el material de aprendizaje suministrado para él. Lo que es más, si los estudiantes tienen diferentes niveles de

conocimiento, necesidades de aprendizaje, equipo y conectividad un sistema recomendador puede ser muy usado en este caso [85].

Los encomender systems no se usarán en el desarrollo de este trabajo de grado, sin embargo se tomará la característica que ellos tienen de dar soporte al perfil tanto software como hardware que posee un estudiante, ya en el contexto del proyecto de grado se pretende usar técnicas explícitas y/o implícitas de identificación de dicho perfil, como las expresadas al inicio de este capítulo en relación con la identificación de la velocidad de conexión, el navegador usado y el sistema operativo.

3.4 DISEÑO DE OBJETOS Y ACTIVIDADES DE APRENDIZAJE [86] [87]

De manera general se muestra ahora una forma de diseñar actividades de aprendizaje dentro de un curso con base en objetos de aprendizaje. Todo curso tiene un ciclo de vida con fases que van desde la planificación hasta la evaluación pasando por la producción e impartición del mismo para finalmente llegar a una fase de análisis de resultados y dependiendo de los resultados de ésta, volver o no a etapas anteriores con propuestas de mejora del curso.

En la fase de planificación es importante saber que no es necesario empezar de cero ya que hay gran cantidad de material existente, de esa forma los esfuerzos pueden concentrarse en la adecuación y mejora del material existente. Análogamente la producción puede convertirse en la agregación y personalización de materiales previos. Si se piensa en la adaptación al estudiante se puede pensar también en varias agrupaciones de las mismas piezas.

En la fase de impartición del curso se pueden tener secuenciaciones distintas del mismo material, así como también el diseño orientado a competencias. En cuanto a la evaluación se puede anotar que es más sencillo determinar las fallas si se tiene separado el contenido de las actividades de aprendizaje. Es importante tener en cuenta que la calidad de los objetos de aprendizaje mejora con el uso.

Una actividad de aprendizaje se diseña teniendo en cuenta el tipo de objetivos del curso, por esa razón puede tratarse de actividades conceptuales donde pesa más el contenido teórico, actividades procedimentales donde pesa más el contenido práctico y actividades actitudinales en las que se hace mayor énfasis en la comunicación, reflexión, análisis y síntesis.

Un recurso educativo es un elemento que permite transmitir conocimiento al aprendiz dentro de una experiencia educativa, que a su vez se define como un conjunto de actividades soportadas por recursos educativos convenientemente organizados. Dentro de un curso los recursos educativos son los contenidos del mismo, así como las actividades son las experiencias educativas.

Para la creación de objetos de aprendizaje se deben atomizar los recursos y actividades en pequeñas piezas autocontenidas, con un objetivo educativo claro y que puedan ser utilizadas en distintos contextos. El paso siguiente es describir los recursos mediante metadatos, para eso se pueden usar estándares como LOM o SCORM [88].

En ese orden de ideas, una forma adecuada de empezar a crear objetos de aprendizaje es comenzando por definir una estructura para todo el curso, para eso se puede hacer una fragmentación lógica redactando las lecciones y temas para cada lección, posteriormente se establece la jerarquía con la que se va a organizar los temas, es recomendable pensar si hay más de una organización posible. Cada tema debe tener una duración y la duración de todos los temas debe ajustarse a la duración del curso, esto se conoce como fragmentación temporal. Una forma de analizar los temas creados es pensar en el tiempo medio que un estudiante dedica al curso en cada clase o sesión de estudio. Si la duración total de todos los temas es mayor que la duración del curso, se debe dividir cada uno de ellos en fragmentos auto-contenidos más pequeños. Cada fragmento de duración limitada se conoce como unidad didáctica. Cada unidad didáctica está compuesta de tareas o unidades mínimas de información educativa, ejemplos de unidades didácticas y tareas pueden verse en la tabla 6 y en la figura 8 respectivamente.

Tarea	Duración (minutos)
Lectura	5-10
Ejercicios	15-20
Caso de estudio	15-20
Proyecto personal	20-30
Otras tareas	35-40
Total	90-120

Tabla 7 Ejemplo de una Unidad Didáctica

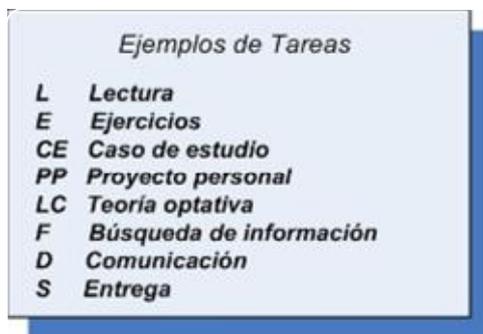


Figura 8 Ejemplo de tareas comunes

En este punto se definen los objetivos específicos de cada unidad didáctica. Posteriormente se decide qué tipo de tareas se va a utilizar para conseguir esos objetivos. Una forma de identificar los objetivos es preguntarse:

- ¿Qué debería saber el estudiante al terminar la clase?
- ¿Qué debería ser capaz de hacer?
- ¿Qué debe haber cambiado en su actitud o escala de valores?

Cada tarea puede formarse de objetos de aprendizaje, que son ficheros construidos con un objetivo educativo y contienen los metadatos o información que describe el curso. Así que en este punto se crea el contenido de cada una de las tareas, teniendo en cuenta que el curso será impartido mediante una plataforma educativa. También es importante establecer un tipo de presentación para cada tipo de tareas de modo que todos los contenidos del curso tengan un aspecto homogéneo.

Ahora es importante tener en cuenta la navegación sabiendo que se pueden crear varios cursos a partir del mismo material, para ello se puede pensar en varias organizaciones posibles del material teniendo en cuenta la organización lógica impuesta por el contenido, la adaptación al perfil del estudiante mediante su estilo de aprendizaje y/o su nivel de conocimiento, y la adaptación al tipo de uso, por ejemplo, se puede tratar de un curso completo, un repaso o una consulta.

El definir la estructura del curso mediante ficheros vacíos puede resultar de gran ayuda, así se puede crear un fichero para cada curso, dentro de éste se puede crear un fichero vacío para cada tema del curso, dentro de cada fichero correspondiente a un tema se puede crear un fichero vacío por cada sesión, finalmente se puede crear un fichero vacío por cada tarea dentro de una sesión. El contenido se pondría en los ficheros de tareas, cada sesión será entonces un índice de tareas, cada tema será un índice de sesiones y cada curso será un índice de temas.

Esta es una forma organizada de desarrollar cursos completos y definir las actividades de aprendizaje necesarias para alcanzar los objetivos propuestos dentro de dichos cursos, todo apoyado en tareas construidas a partir de objetos de aprendizaje. En la siguiente sección se muestran algunos lineamientos relacionados con la creación de contenidos educativos, en el capítulo 4 se muestra la implementación de una herramienta Web que puede facilitar dicha creación, así como también la configuración de un entorno integrado de desarrollo que posibilite un proceso eficiente de generación de contenidos siguiendo diferentes rutas de aprendizaje.

3.5 GUÍA METODOLÓGICA PARA LA CONSTRUCCIÓN DE CONTENIDOS Y RUTAS DE APRENDIZAJE ADAPTADOS AL PERFIL DEL ESTUDIANTE

Para que la generación de contenidos a nivel Web sea un proceso efectivo se debe considerar algunos aspectos tanto a nivel pedagógico como técnico, cabe aclarar que el aspecto pedagógico resulta muy complejo para el equipo de desarrollo y la ayuda especializada de terceros está limitada al trabajo desarrollado por la doctora Mary Carmen Fernández Panadero [86] [87], y referencias a trabajos encontrados en Internet.

En dotLRN no se cuenta con una herramienta para la construcción de objetos de aprendizaje, sin embargo LORS [89] permite el manejo de assets³² y SCOs³³ una vez éstos hayan sido creados. Por ello es importante tener en cuenta algunos lineamientos fundamentales para el desarrollo de una herramienta Web que brinde soporte a la creación de assets y/o SCOs. Además, el desarrollo de objetos de aprendizaje (SCOs) y de assets debe organizarse de tal manera que, para el caso de E-LANE en la Universidad del Cauca, posibilite una fácil y rápida utilización y aprovechamiento de la herramienta implementada. Adicionalmente, los profesores que desarrollen assets y/o SCOs deben seguir algunos lineamientos pedagógicos, a manera de guía metodológica, que permitan la creación de rutas de aprendizaje para los estudiantes.

Con el fin de alcanzar un máximo aprovechamiento de los lineamientos propuestos se decidió separarlos de la siguiente manera:

-Lineamientos para la generación de contenidos y rutas de aprendizaje personalizados.

-Lineamientos para la construcción de una herramienta Web con soporte para la generación de contenidos de aprendizaje.

3.5.1 Lineamientos para la construcción de contenidos y rutas de aprendizaje personalizados

Guía metodológica: Se pretende definir una guía metodológica para la generación de contenidos de aprendizaje tan variados que brinden soporte a distintas rutas de aprendizaje las cuales deben ser establecidas en función de los intereses, necesidades formativas y estilos preferidos de cada estudiante. A continuación se presentan las pautas que debería seguir el profesor para la creación adecuada de un curso:

Recomendaciones Pedagógicas

³² **Asset:** Representación electrónica de multimedia, texto, imágenes, sonido, páginas Web, piezas de datos en general.

³³ **SCO:** Sharable Content Object.

Antes de definir una guía metodológica clara, es importante tener en cuenta las siguientes recomendaciones generales a manera de introducción:

- Para que el proceso de enseñanza-aprendizaje mediado por Internet se realice eficazmente es importante identificar con anterioridad qué, cómo y cuando se va a enseñar, además de otros factores que en conjunto forman parte de la planeación y el desarrollo del contenido del curso en línea.
- Analizar el tipo de material o contenido a desarrollar con el fin de identificar cual sería la mejor forma de organizar la información. Dependiendo del tema a desarrollar, el profesor debe establecer un orden apropiado para la información.
- Definir los objetivos para el curso.
- Seguir una estructura jerárquica de composición de los cursos (tema, asignatura, curso, unidad didáctica, tarea).
- Desarrollar el tema de tal forma que cada párrafo contenga una sola idea con el fin de evitar una saturación que genere indisposición en el estudiante durante el proceso de lectura.
- Utilizar estilo de redacción en pirámide invertida, es decir, comenzando por la conclusión, síntesis o idea general del tema y siguiendo con el desarrollo de los detalles del mismo. De esta forma el estudiante tendrá una idea global de lo que va a leer [90].
- Puesto que los contenidos son de carácter educativo, se debe utilizar un lenguaje objetivo y simple, que posibilite un proceso de lectura más rápido, sin emplear palabras redundantes o afirmaciones no basadas en evidencias tal como ocurre con el lenguaje promocional.

Recomendaciones Tecnológicas:

En el aprendizaje en línea resulta fundamental tener en cuenta los recursos tecnológicos disponibles por parte del estudiante, por ello se recomienda:

- Considerar la infraestructura tecnológica, especialmente la capacidad de conexión y la disponibilidad de conectividad del estudiante para poder adaptar los contenidos desarrollados por el profesor a las condiciones de acceso de cada estudiante. De este

modo si se trata de video streaming, se deben utilizar archivos más livianos de acuerdo a la velocidad de conexión. La idea es brindar cursos de calidad independientemente de las condiciones de acceso de los estudiantes. No se trata de quitar recursos valiosos sino de optimizarlos.

- En relación con el contenido, es recomendable utilizar múltiples formatos (texto, imágenes, sonidos y videos) y múltiples calidades de acuerdo al tipo de conectividad para ofrecer una variedad de elementos adecuada a los diferentes estilos de aprendizaje.
- Los contenidos creados deben ser ricos en texto plano e imágenes livianas descartándose formatos no comprimidos como BMP para ser visualizados por los estudiantes que disponen de una baja conectividad de tal forma que incluso los recursos de video de baja calidad no sean soportados.
- Garantizar la disponibilidad de la actividad e-learning, lo que se traduce en instalaciones redundantes. Replicar los nodos de información para que frente a cualquier eventualidad la información esté disponible al usuario (estudiante). [99]
- Garantizar el rendimiento de los equipos informáticos (servidores) en donde se realicen peticiones que requieren de recursos de alta capacidad como en el caso de transformación de documentos xml a html. [99]

Usabilidad

Con base en [75] [77] [79] [84] [90] [98] se establecen algunos principios básicos a considerar para el desarrollo de contenidos destinados a la Web:

- Tener en cuenta la accesibilidad, navegabilidad, optimización y productividad del contenido, especialmente si está constituido de páginas html.
- Emplear un estilo de redacción apropiado a la documentación Web con el fin de mejorar la experiencia del usuario en el uso de los contenidos creados. Los usuarios en la Web por lo general no realizan una lectura profunda sino que escanean la información.

- Incluir hipervínculos hacia sitios Web que contengan información complementaria al tema que se está tratando.
- Utilizar colores de forma adecuada de acuerdo a lo indicado en el apartado 3.1.3 (sobre usabilidad de los contenidos) de este capítulo.
- Tener en cuenta la distribución del contenido en la pantalla, es decir, el tamaño o ancho del logotipo superior, el ancho de la ventana de navegación inferior, las zonas de información, etc.

En el capítulo cinco se desarrolla un curso utilizando como base los lineamientos planteados en el presente capítulo y herramientas de soporte adecuadas; para el presente documento se utiliza la herramienta RACE-Editor, sin embargo, no se descarta el uso de herramientas de escritorio, razón por la que se tiene una recomendación más:

- Utilizar herramientas Web y de escritorio preferiblemente software libre para el desarrollo de contenidos de aprendizaje, como RACE, XMLmind, Reload Editor (en sus versiones Web y de escritorio), entre otras.

Con base en las recomendaciones pedagógicas, tecnológicas y de usabilidad, se sugiere utilizar la siguiente **Guía metodológica** [86] [87] para la generación de contenidos educativos:

1. Comience siempre definiendo los objetivos del curso.
2. Es recomendable crear la estructura antes de empezar a trabajar sobre el contenido (fragmentación lógica). Se pueden redactar las lecciones con sus temas y establecer la jerarquía con la que se van a organizar dichos temas.
3. Tenga en cuenta la duración de las clases o sesiones de trabajo (fragmentación temporal). Se trata de asignar una duración a cada tema y ajustarlos a la duración del

curso. Se puede pensar en el tiempo medio que un estudiante dedica a cada curso en cada sesión (clase) del curso.

4. Se debe establecer un diseño de presentación antes de empezar a trabajar (fragmentación espacial), es decir definir en que forma homogénea se presentarán los contenidos de cada tarea.
5. Tener en cuenta cada uno de los estilos de aprendizaje (ver anexo B), ya que la oferta de formación no va a estar destinada exclusivamente a estudiantes que asimilen los conocimientos de una forma determinada y prefieran unos determinados recursos para aprender. Por lo tanto se debe definir hacia qué perfil o perfiles de estudiante irá dirigido el objeto de aprendizaje a crear.
6. Establecer los objetivos específicos para cada unidad didáctica (tema específico con un tiempo definido). Se puede preguntar: ¿Que debería saber el estudiante al terminar esta clase? ¿Qué debería ser capaz de hacer? ¿Qué debe haber cambiado en su actitud o escala de valores?, esto lleva a definir objetivos conceptuales, procedimentales y actitudinales respectivamente.
7. Decidir que tipo de tareas (unidades mínimas de información educativa) se va a utilizar para conseguir esos objetivos. Las tareas pueden ser lecturas, ejercicios, caso de estudio, proyecto personal u otras tareas.
8. Crear el contenido de las tareas, se debe tener en cuenta que no es necesario empezar desde cero ya que hay gran cantidad de material existente.
9. Agregar metadatos que describan los recursos usando especificaciones tendientes a estándares educativos como SCORM o LOM.
10. Permitir revisión y comentarios por parte de un grupo de expertos antes de dar por terminado un contenido [91].
11. Publicar los contenidos en un repositorio central.
12. Analizar el resultado de los contenidos.

En el capítulo siguiente se mostrará la implementación de una herramienta que ayudará en gran medida al profesor en el desarrollo de un curso, facilitándole la creación de contenidos educativos.

Al seguir una guía metodológica como la planteada, no se pretende en ningún momento crear una “camisa de fuerza”, al contrario, se trata de dotar a los profesores con los parámetros y herramientas adecuadas para desarrollar cursos completos en base a contenidos de aprendizaje constituidos por múltiples recursos, ofreciendo al mismo tiempo la posibilidad de que los estudiantes puedan seguir distintas rutas de aprendizaje de acuerdo a sus preferencias y condiciones técnicas para el acceso a dichos cursos.

3.5.2 Lineamientos para la construcción de una herramienta Web con soporte para la generación de contenidos de aprendizaje

Con base en los lineamientos para generación de contenidos y rutas de aprendizaje personalizados incluidos en el numeral anterior se plantea las características que debería cumplir una herramienta capaz de brindar soporte para la creación de contenidos dentro del LMS dotLRN.

- 1.) El proceso actual de creación de contenidos de aprendizaje tiene como base xml y específicamente la DTD de docBook, a partir de los contenidos creados de esta forma, se realizan transformaciones adecuadas mediante el uso de motores XSLT y hojas de estilo XSL personalizadas. Con el desarrollo de una herramienta Web para la generación de contenidos se busca que el proceso anterior sea automático y se realice de forma transparente al usuario.
- 2.) Para personalizar los contenidos creados, la herramienta debe ser capaz de utilizar las etiquetas profile de docbook y aplicar de igual forma hojas de estilo CSS adecuadas.
- 3.) El manejo de permisos resulta primordial en aplicaciones como la que se va a desarrollar, esto permite acceso únicamente a determinados usuarios y bajo ciertas políticas de uso, de esta forma un profesor podrá escribir borradores de contenidos educativos, visualizarlos, editarlos y borrarlos; mientras que un tutor o colaborador del profesor sólo podrá leer el contenido y agregar comentarios sobre el mismo, mas no editarlo.
- 4.) La herramienta a construir debe ser capaz de incorporar las especificaciones IMS Metadata y Content Packaging así como las extensiones de ADL SCORM, esto dado que LORS incorpora este tipo de especificaciones y extensiones.

- 5.) LORSm³⁴ reúne todas las funciones del LORS para importar, exportar, gestionar y entregar paquetes de contenido IMS, Metadatos IMS y paquetes de conformidad con SCORM, esta aplicación debe ser compatible con la herramienta que se está desarrollando, permitiendo que los contenidos generados usen las funciones de LORS y sean accesibles desde LORSm.
- 6.) Se debe crear un Applet y un Portlet para permitir el acceso de un profesor a la herramienta desde una determinada clase.
- 7.) Dado que LORS no incorpora herramientas de autoría, la herramienta a desarrollar debe cubrir, al menos en parte, algunas de las facilidades ofrecidas por Reload Editor, que es la herramienta utilizada actualmente en la Universidad del Cauca para empaquetar y crear metadatos, ésta herramienta soporta las extensiones de IMS y SCORM. Esto implica que la herramienta debe empaquetar contenidos de acuerdo a ADL SCORM 1.2 o versiones superiores, para ello basta con crear el *imsmanifest.xml* adecuado, crear metadatos para el objeto de información, adicionar organizaciones, ítems y subítems y adicionar los recursos.
- 8.) LORS utiliza la aplicación file storage para almacenar y publicar los contenidos, file storage a su vez está construida en base al repositorio de contenidos de OpenACS, por lo tanto la herramienta a implementar debe igualmente almacenar los contenidos en el file storage incluyendo el *imsmanifest.xml* y los metadatos correspondientes, de tal forma que permita su exportación como paquete SCORM.
- 9.) Dado que se utilizará el repositorio de contenidos, se puede usar de igual forma toda su funcionalidad (o al menos extenderla) como lo hace la aplicación para creación de presentaciones Wimpy Point, esta funcionalidad incluye: manejo de permisos, adición de imágenes de manera flexible, edición de una presentación y/o slide cualquiera, visualización de las presentaciones creadas, eliminación de una presentación y/o slide cualquiera (si se cuenta con los permisos para borrar), selección y/o creación de una CSS para la presentación y soporte para el manejo de versiones de un mismo contenido.

Estas características permiten abstraer los siguientes lineamientos, útiles al momento de construir una herramienta de edición de contenidos para la Web:

³⁴ **LORSm**: Aplicación de gestión para LORS o Servicio de Repositorio de Objetos de Aprendizaje.

- Los contenidos creados con la herramienta deben estar descritos mediante xml, esto les brinda mayor flexibilidad, brindando la posibilidad incluso para la creación de metamodelos de cursos. Además, todas las especificaciones³⁵, normas y estándares tecnológicos actuales usan XML como medio universal para describir, transformar y transportar datos entre los distintos LMS.
- La interfaz con el usuario puede ser WYSIWYG o WYSIWYM, esto le facilita enormemente el manejo de la herramienta.
- Es recomendable que este tipo de herramientas creen contenidos basados en XML, y que además utilicen toda la potencialidad que brinda el lenguaje de marcado extensible, esto en lo relacionado a schemas (DTD, Schema, Relax-NG), hojas de estilo (XSL-FO, XSLT, Xlink, Xpointer, CSS), y programación (DOM, JDOM, SAX, STAX).
- Se debe tener en cuenta la importancia de contar con un sistema de permisos para poder controlar de manera fácil el acceso a la aplicación para la creación de contenidos.
- Se debe trabajar bajo especificaciones tendientes a la interoperabilidad como IMS y SCORM. En este caso, dependiendo de la especificación se deben implementar módulos adicionales como por el ejemplo un Runtime Environment para el caso de manejo de SCO's.
- La aplicación debe ser capaz de guardar borradores de contenidos preferiblemente en XML por las ventajas ya mencionadas, además resulta importante el enfoque hacia el desarrollo de una herramienta colaborativa, mediante la cual varios usuarios puedan crear un contenido, hacer comentarios y realizar revisiones al mismo.
- La aplicación debe permitir la edición de texto y adición de diferentes tipos de recursos como audio, video, animaciones, etc.
- Finalmente la aplicación debe brindar una previsualización de los contenidos cuando aún se encuentren en estado de borrador. Además se debe dar la facilidad de exportar los contenidos creados y empaquetarlos de acuerdo a especificaciones destinadas a tal fin tales como SCORM.
- Se debe tener en cuenta la accesibilidad y usabilidad de los contenidos que se creen mediante la herramienta a desarrollar.

³⁵ Para mayor información sobre las especificaciones, remitirse al Capítulo 2, numeral 2.1. Los estándares en el proceso de e-learning.

Es importante aclarar que este último lineamiento pretende identificar la accesibilidad y la usabilidad como aspectos importantes en la creación de contenidos educativos, mas no pretende reducir o encapsular estas dos grandes áreas del conocimiento, al contrario se deja una puerta abierta hacia dichas áreas del conocimiento.³⁶

4 ANÁLISIS DE UN ENTORNO INTEGRADO DE DESARROLLO PARA EL PROYECTO E-LANE EN LA UNIVERSIDAD DEL CAUCA

La adaptabilidad de los contenidos es un aspecto que debe ser implementado en cualquier sistema de gestión de aprendizaje, para el caso de este proyecto se diseña e implementa un sistema para generación de contenidos teniendo en cuenta el perfil del estudiante. Como se mencionó en el capítulo 3, dicho perfil está constituido principalmente por el estilo de aprendizaje y las condiciones de conectividad del aprendiz. A partir de estos dos aspectos se pretende brindar soporte a la adaptabilidad al interior del LMS dotLRN. Un entorno integrado de desarrollo debe ofrecer diferentes herramientas que posibiliten un proceso de enseñanza-aprendizaje más efectivo, por lo tanto es importante considerar, en este caso para dotLRN, además de RACE Editor, otras aplicaciones suministradas por el LMS que pueden ser usadas en clases y comunidades y ofrecen diferentes funcionalidades como complemento al proceso educativo, de este modo se tendrá en cuenta las aplicaciones que de alguna manera puedan brindar soporte para la puesta en práctica de los lineamientos planteados en el capítulo 3.

4.1 CONFIGURACIÓN DEL ENTORNO VIRTUAL DE APRENDIZAJE

³⁶ Se puede encontrar mayor información en [98]

El entorno virtual de aprendizaje dotLRN está constituido por una gran cantidad de paquetes que ofrecen aplicaciones con diferentes funcionalidades que pueden ser de gran utilidad para los actores del proceso educativo. También existen paquetes de OpenACS como assessment, survey y evaluation que pueden ser integrados a dotLRN y utilizados con fines similares.

A continuación se realiza una descripción de cada uno de dichos paquetes:

- *Assessment y Survey:* Estos paquetes permiten la realización de cuestionarios, tests y exámenes cuyas respuestas son procesadas automáticamente y almacenadas en un sistema de control de notas. Estos módulos podrían ser utilizados por el profesor para evaluar el contenido de cada tema desarrollado con RACE Editor. Un test puede constituir una tarea de aprendizaje que en conjunto con los elementos y recursos del contenido forman una unidad didáctica con lo que se cumple con las recomendaciones en cuanto a estructuración de los contenidos formuladas en [84] [85] y se brinda soporte al lineamiento sobre estructura de composición de los cursos (tarea, unidad didáctica, tema, curso y asignatura) formulado en el numeral 3.5.2 del presente documento.
- *Evaluation:* Este paquete permite a los profesores asignar tareas a los estudiantes y calificarlas fijando porcentajes de nota a cada una de las tareas de manera similar a como se realiza en el sistema educativo convencional. Los estudiantes tienen la posibilidad de subir la solución a cada tarea y posteriormente ver los comentarios del profesor sobre dicha solución y la calificación obtenida. De acuerdo al porcentaje asignado a cada tarea, el estudiante puede conocer sus notas parciales y al final su nota definitiva. Con el uso de este paquete se podría dar soporte al lineamiento sobre evaluación como fase final de la creación de objetos de aprendizaje enunciado en la metodología formulada en el capítulo 3.
- *Curriculum:* Este módulo permite al profesor colocar URLs de recursos de aprendizaje sugeridos en una serie de elementos educativos que sirven de guía para abordar el material de aprendizaje de un determinado curso, un camino de estudio que los estudiantes son invitados a seguir. Este módulo puede dar soporte al lineamiento sobre la inserción de hipervínculos hacia sitios Web que contengan información complementaria al tema que se está tratando.

- *dotLRN Homework*: Este paquete provee un buzón para los estudiantes con el fin de que puedan subir archivos de tareas y para los profesores un medio para que puedan subir sus comentarios y correcciones sobre tales archivos. Este módulo puede dar soporte al lineamiento sobre el tipo de tareas a usar (lecturas, ejercicios, proyectos, etc.) y el seguimiento que se debe hacer a dichas tareas y trabajos. De igual forma puede dar soporte al lineamiento sobre la adición de revisiones y comentarios por parte de los conoedores de cada temática sobre un contenido.
- *Syllabus*: Permite al usuario tener siempre disponible el programa de estudios de las clases o comunidades de dotLRN, este programa puede estar en un archivo que debe ser subido a la plataforma o se puede referenciar mediante un hipervínculo a una localización específica en la Web. Este módulo puede dar soporte al lineamiento sobre el establecimiento de una organización adecuada de la información que facilite al estudiante el proceso de aprendizaje.
- *Glossary*: Este paquete soporta glosarios para múltiples contextos. Un sitio, un subsitio, un grupo, un usuario o incluso un documento puede tener uno o más glosarios asociados con éste. El contenido de un glosario es guardado en el repositorio de contenidos. Se podría brindar soporte al lineamiento sobre creación de un glosario de acrónimos y palabras técnicas que esclarezcan su significado en el contexto del contenido creado.
- *IMS LD*: IMS Learning Design es una descripción basada en XML para e-learning. Permite a los profesores establecer un flujo de trabajo en cada uno de sus cursos especificando un conjunto de actividades de aprendizaje (las cuales se refieren usualmente a un conjunto de recursos) y quién y cuándo pueden desarrollarse estas actividades y bajo qué condiciones. Por lo tanto, establece una secuencia de actividades para cada rol. En IMS-LD se utiliza el término “unidad de aprendizaje” para referirse a una lección, un módulo o un curso, es decir, a una colección de actividades de aprendizaje ordenadas las cuales tienen asociados recursos y servicios. Este módulo brinda soporte a los lineamientos de descripción de los contenidos creados mediante XML para brindar flexibilidad y utilización de especificaciones tendientes a la interoperabilidad como las formuladas por IMS-SCORM. También puede brindar soporte

al lineamiento sobre definición de los objetivos para un curso ya que con IMS-LD se puede asignar metas a cada unidad de aprendizaje que deben ser cumplidas por el estudiante y finalmente permite organizar adecuadamente la información al establecer una secuenciación apropiada de los contenidos.

- *LAMS*: Es una herramienta para el diseño, gestión y despliegue de actividades de aprendizaje colaborativo en línea. Provee un ambiente visual altamente intuitivo para la creación de secuencias de actividades de aprendizaje. El profesor puede incluir diferentes actividades tales como lecturas y contenido general en formato HTML, preguntas y respuestas, intercambio de recursos, foros de discusión, etc. Dentro de cada actividad, el estudiante tiene la posibilidad de tomar apuntes e incluir observaciones sobre el tema de estudio, además, las respuestas de cada estudiante a las preguntas formuladas por el profesor pueden ser observadas por los demás estudiantes como medio de colaboración. Esta herramienta es de gran utilidad ya que permite la creación a nivel Web de actividades de aprendizaje, aunque el profesor puede definir una secuencia determinada para el contenido estableciendo una ruta de aprendizaje sobre el mismo, esta herramienta a diferencia de RACE no ofrece personalización de contenidos. LAMS constituye una herramienta Wysiwyg que puede brindar soporte a los lineamientos sobre creación de actividades de aprendizaje, utilización de herramientas colaborativas y soporte para múltiples recursos como imágenes, video, audio y animaciones.
- *Beehive*: Es un framework de aplicaciones Web para el diseño y soporte de aprendizaje colaborativo sincrónico. Simplifica el desarrollo de herramientas colaborativas requeridas para implementar estrategias de diseño educativo. Guía a los profesores en la implementación de actividades en línea basadas en técnicas pedagógicas bien conocidas para lograr sus objetivos educativos. En el portal de administración de cada curso en dotLRN, los profesores pueden acceder al administrador de beehive para agregar nuevas actividades al curso, cambiar o borrar las existentes. El gestor de beehive incluye tres componentes principales: la página de información de sesión en la que los profesores pueden llenar la información relacionada con la sesión como título, fecha, descripción, etc.; la página de tareas de sesión en la cual ellos pueden seleccionar las tareas requeridas para dicha sesión; y la página de estructura de sesión donde pueden

secuenciar las tareas seleccionadas de acuerdo al escenario de la actividad. Este módulo podría ofrecer soporte a los mismos lineamientos considerados por LAMS.

Hasta este punto se ha realizado una breve descripción de algunos de los paquetes de la plataforma dotLRN/OpenACS que pueden ayudar al profesor en el proceso de generación de contenidos teniendo en cuenta los lineamientos establecidos en el capítulo anterior de este documento. No se realizó una descripción detallada de las aplicaciones tratadas anteriormente puesto que su estado de desarrollo es maduro a excepción de IMS-LD y LAMS, por ello existen manuales didácticos al respecto [92]. Ahora se describe el proceso de análisis, diseño e implementación del sistema RACE.

4.2 SISTEMA RACE (Rutas de Aprendizaje sobre Contenidos Educativos)

El sistema RACE es un conjunto de aplicaciones y servicios desarrollado sobre la plataforma dotLRN/OpenACS con el fin de proveer un mecanismo para la producción, ensamblaje y publicación de contenidos educativos (objetos de aprendizaje) teniendo en cuenta el perfil del estudiante. Este sistema ofrece personalización de los contenidos, desde el punto de vista de la presentación de los mismos, de acuerdo a las preferencias de aprendizaje y condiciones técnicas del estudiante. A nivel funcional es un prototipo que cumple adecuadamente los objetivos que se plantearon en la fase de formulación del presente proyecto. A continuación se presenta una vista general del sistema y los elementos que interactúan con él:

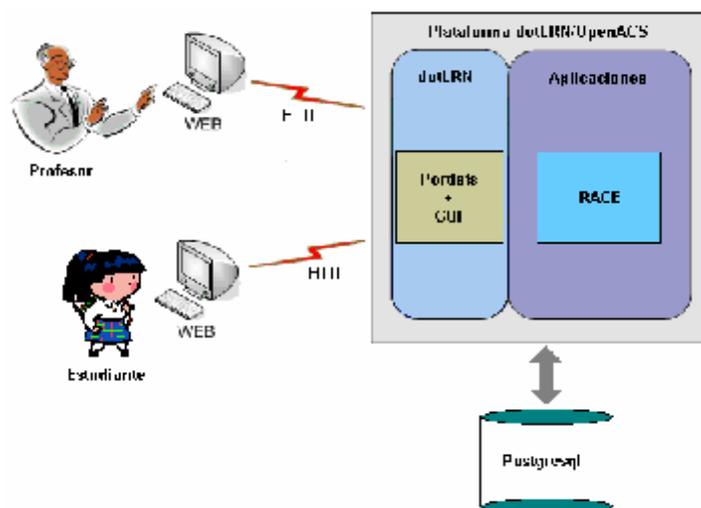


Figura 9 Vista general del sistema y su entorno

De la figura anterior cabe resaltar que toda la interacción usuario/sistema se realiza vía Web a través del navegador.

- *Usuario*: Hay dos roles de usuario, el profesor y el estudiante. El primero se encarga de crear objetos de aprendizaje teniendo en cuenta el perfil del estudiante. El segundo es quien recibe el contenido de los objetos de forma personalizada según sus preferencias y condición técnica (el tipo de conexión).
- *Plataforma dotLRN/OpenACS*: Es la estructura software sobre la cual está implementado el sistema.
- *RACE*: El sistema *Rutas de Aprendizaje sobre Contenidos Educativos* constituye el conjunto de aplicaciones y servicios implementados. En la figura 10 se muestra los componentes del sistema.
- *Postgresql*: Es el RDBMS utilizado por la plataforma y por lo tanto por el sistema RACE.

4.2.1 Arquitectura

Por razones de análisis y diseño, el sistema fué dividido en cuatro módulos o subsistemas: Race Editor, Race Profile, Race Content y Race Agent. A continuación se muestra la arquitectura del sistema y posteriormente se realiza una descripción de sus componentes.

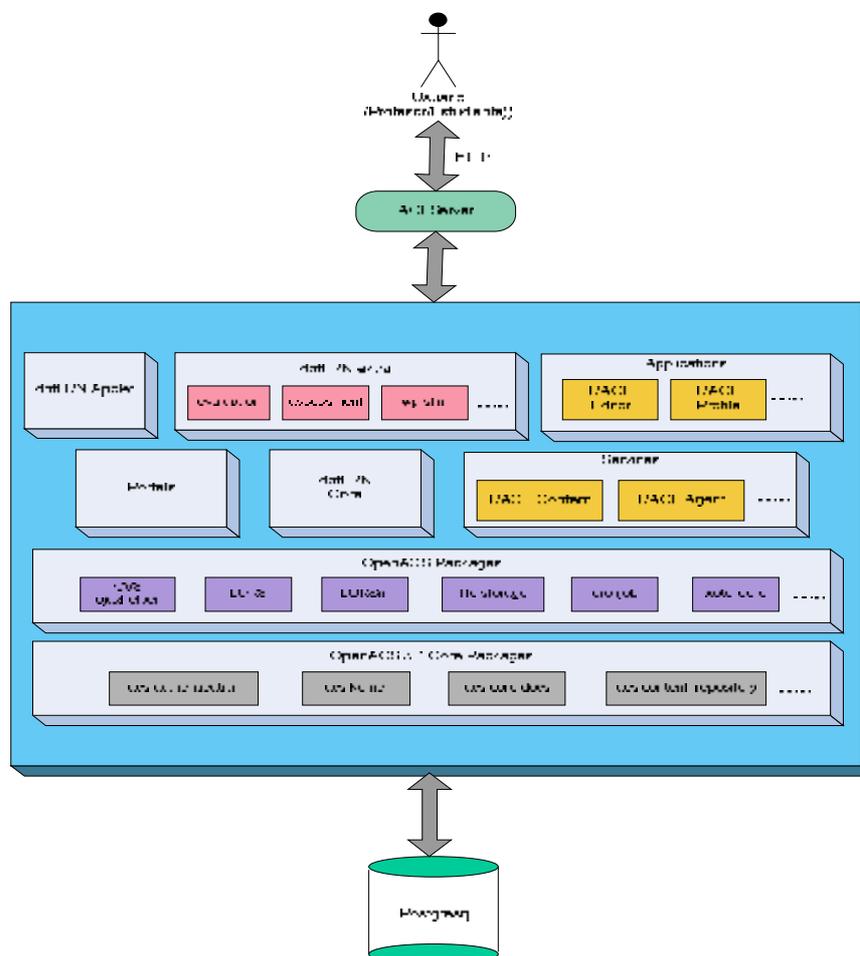


Figura 10 Arquitectura del sistema a implementar

- *RACE Editor*: Es un subsistema para la generación de objetos de aprendizaje, utilizada por el profesor para crear contenido educativo personalizado de acuerdo al perfil del estudiante. Brinda soporte para la inserción de texto, imágenes y otro tipo de recursos como video y animaciones. La herramienta cumple con las recomendaciones para la construcción de una herramienta Web con soporte para objetos de aprendizaje formulados en el capítulo 3.
- *RACE Profile*: Es un subsistema diseñado para ser utilizado por el estudiante. Incluye dos funcionalidades, una que le permite conocer su estilo de aprendizaje a través de la solución del test de estilos de aprendizaje de David Kolb, la otra funcionalidad permite

determinar la velocidad de conexión del estudiante. Con estos dos parámetros el sistema reconoce que tipo de contenido será mostrado al estudiante.

- *RACE Content*: Este subsistema está conformado por los procedimientos necesarios para el procesamiento de los contenidos creados por el profesor con RACE Editor, está apoyado con scripts en bash y perl que posibilitan el procesamiento y las transformaciones adecuadas del contenido con las hojas de estilo de Norman Walsh adaptadas por el equipo E-LANE en la Universidad Carlos III de Madrid para la creación de contenidos educativos. Además, dichos scripts también permiten el empaquetamiento de los contenidos de acuerdo a SCORM.
- *RACE Agent*: Es un subsistema que posee la lógica de control necesaria para la interacción entre los diferentes módulos del sistema y el usuario. Permite la creación asincrónica de los contenidos entregados por RACE Editor, de acuerdo al nivel de carga del sistema en donde se está ejecutando la plataforma OpenACS, así como también reconoce hacia qué perfil está dirigido dicho contenido, además de desplegar en el repositorio de contenidos el nuevo material de aprendizaje creado. RACE Agent es el encargado de definir el tipo de contenido que se va a mostrar al estudiante de acuerdo a los datos suministrados por RACE Profile.

El resto de los elementos mostrados en la figura 10 son algunos de los paquetes de la plataforma utilizados por el sistema RACE para llevar a cabo las diferentes funciones que desempeña.

4.3 PROCESO DE DESARROLLO

El sistema RACE se desarrolló de acuerdo al procedimiento sugerido en el documento “*Referencia Metodológica Integral para Desarrollo de Sistemas Telemáticos*” [93]. Esta metodología ha sido pensada desde la esencia misma de la Ingeniería Telemática y constituye una herramienta útil para el establecimiento de procesos adecuados de mejoramiento de la calidad de los procesos de desarrollo que buscan crear soluciones de calidad, oportunas y a costos competitivos.

El lenguaje de modelado a utilizar en el proceso de desarrollo es UML (Lenguaje de Modelado Unificado), incluyendo la extensión WAE (Web Application Extensión [94]) de Conallen, la cual extiende UML con estereotipos para permitir el modelado de elementos de la arquitectura específicos de la Web como parte del modelo del sistema. En el desarrollo de RACE se elaboraron los modelos de la Organización, Casos de Uso, Análisis, Diseño e Implementación.

Como se indicó anteriormente, el sistema se dividió en cuatro módulos o subsistemas: Race Editor, Race Profile, Race Content y Race Agent. A continuación se muestra la información más relevante de los modelos elaborados. Para más detalle referirse al anexo A de este documento.

4.3.1 Modelo de la Organización

La organización en la cual se va a implantar el sistema a desarrollar está conformada por los participantes en el proyecto E-LANE en la Universidad del Cauca. El personal que forma parte de esta organización, desde el punto de vista académico, está conformado por docentes, monitores y estudiantes de cada uno de los cursos ofrecidos a través de la plataforma de teleformación “EVA” (dotLRN). Mediante la incorporación de “EVA” en la Universidad del Cauca se brinda una alternativa a los procesos de formación (enseñanza/aprendizaje) convencionales con opciones para la producción y publicación de material académico de buena calidad. A continuación se muestra una descripción de cómo se realiza actualmente el proceso de creación de material educativo al interior de la organización. La tabla 8 muestra la estructura de la organización y el proceso de creación, adecuación y publicación de contenido en el Sistema de Gestión de Aprendizaje dotLRN:

Actores del Negocio	Función dentro de la Organización
<i>Profesor.</i>	Persona que crea los contenidos educativos mediante el uso de herramientas como Microsoft Office, OpenOffice, etc.
<i>Estudiante.</i>	Persona que a través del Web accede al LMS dotLRN con el fin de utilizar y asimilar los contenidos que el profesor a desarrollado para su proceso de formación académica.
<i>Monitor.</i>	Persona que se encarga de cambiar los contenidos a un formato apropiado al Web (docbook y luego HTML) y de aplicar las especificaciones de metadatos y empaquetamiento de SCORM para que el contenido pueda ser publicado en cualquier LMS.
<i>Administrador.</i>	Persona que administra la plataforma dotLRN, es el encargado de

Lineamientos para Generación de Contenidos Educativos y Rutas de Aprendizaje Personalizados para el Proyecto E-LANE en la Universidad del Cauca

	publicar el paquete de contenido en la plataforma.
Trabajadores del Negocio ³⁷	Función dentro de la Organización
<i>Herramientas de Oficina.</i>	Posibilitar la creación de contenidos educativos por parte del profesor.
<i>XMLmind.</i>	Permitir la creación de los contenidos en formato docbook-xml.
<i>Xsltproc.</i>	Procesar y transformar los contenidos docbook en formato HTML mediante la utilización de las hojas de estilo docbook-xsl de Norman Walsh.
<i>Reload.</i>	Generar el archivo estándar <i>imsmanifest.xml</i> con la descripción de la organización y recursos del contenido. Además, permitir la adición de metadatos y empaquetar el contenido incluyendo todos sus recursos (.ZIP)
<i>dotLRN/OpenACS.</i>	Posibilitar el proceso de enseñanza y aprendizaje, permitiendo la publicación y utilización de contenidos educativos.
<i>Navegador.</i>	Hace posible el acceso y la navegación del cliente por los contenidos.
Entidades del Negocio	Descripción
<i>Contenido formato no Web.</i>	Es el contenido producido por el profesor mediante herramientas de oficina. Este contenido tiene un formato determinado dependiendo de la herramienta utilizada en su creación (.doc, por ejemplo), pero este formato no es apropiado para ser publicado a través del Web.
<i>Contenido docbook.</i>	Es el contenido generado a partir de la herramienta XMLmind. El formato de este contenido es .xml
<i>Contenido html.</i>	Es el que se genera de la transformación de los archivos docbook mediante XSL, utilizando el procesador xsltproc. Como su nombre lo indica, son archivos de formato .html, el cual es el formato más común para utilizarse a través del Web.
<i>Paquete SCORM.</i>	Es un archivo .ZIP que incluye todo el contenido procesado de acuerdo a SCORM.
<i>Contenido publicado.</i>	Cuando se publica el paquete, este se descomprime en la plataforma y todos los archivos que constituyen el contenido se organizan y quedan disponibles para ser utilizados por el estudiante.

Tabla 8 Estructura de la Organización

A continuación se presenta el diagrama de Casos de Uso de la Organización y una breve descripción de cada caso.

³⁷ Para el presente proyecto, asumiendo que la organización es el entorno integrado para la producción de contenidos de aprendizaje, se modelan como Trabajadores del Negocio las herramientas que utiliza el proyecto E-LANE en el sistema actual en la Universidad del Cauca.

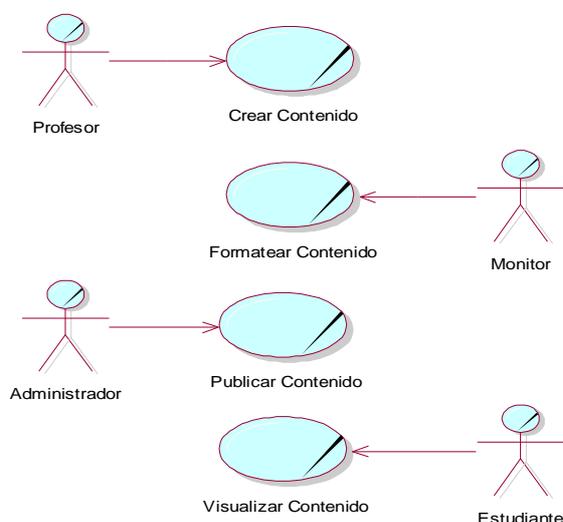


Figura 11 Diagrama de Casos de Uso de la Organización

Descripción de los casos de uso de la Organización:

Caso de uso:	Crear Contenido
Actores:	Profesor (iniciador)
Descripción:	El profesor crea sus contenidos con herramientas de Oficina como Office, OpenOffice, etc. Este contenido no tiene el formato apropiado para ser colocado en la plataforma dotLRN para ser visualizado a través de Internet.
Caso de uso:	Formatear Contenido
Actores:	Monitor(iniciador)
Descripción:	El Monitor toma los contenidos creados por el profesor y utiliza la herramienta XMLmind para generarlos en formato docbook, posteriormente utiliza xsltproc para generar salidas HTML equivalentes del contenido. Finalmente utiliza la herramienta Reload para aplicar las especificaciones de metadatos y empaquetamiento de SCORM para posibilitar la reutilización del contenido en cualquier LMS que soporte estas especificaciones. En este punto el contenido está empaquetado y es apto para ser publicado en la plataforma.
Caso de uso:	Publicar Contenido
Actores:	Administrador (iniciador)
Descripción:	Una vez el Monitor ha generado el paquete de contenido, el Administrador toma este paquete y lo sube a la plataforma para ser publicado en el repositorio de contenidos de dotLRN/OpenACS.
Caso de uso:	Visualizar Contenido
Actores:	Estudiante (iniciador)
Descripción:	Una vez el contenido ha sido publicado en dotLRN, el estudiante accede al curso(s), al cual ha sido inscrito previamente, a través de su navegador para ver y utilizar los contenidos académicos.

4.3.2 Modelo de Casos de Uso del Sistema

A continuación se presenta el Diagrama de Casos de Uso del sistema RACE. En vista de la magnitud del sistema, los diagramas de Casos de Uso de diseño de cada uno de los subsistemas que conforman RACE han sido incluidos en el anexo A de la documentación.

Diagrama de Casos de Uso del Sistema RACE

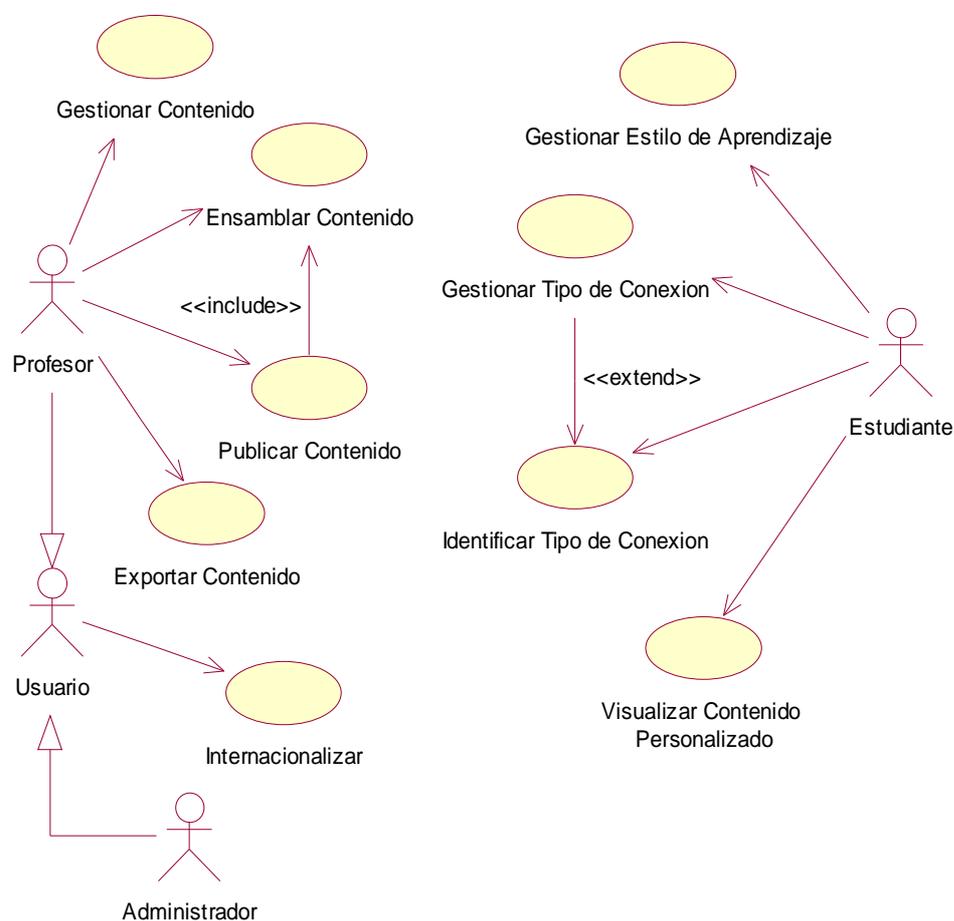


Figura 12 Diagrama de Casos de Uso del sistema RACE

De la figura anterior se especifica a continuación como está constituido cada uno de los módulos que componen el sistema completo:

- **Subsistema 1: Aplicación RACE Editor**

Constituido por los siguientes Casos de Uso:

- Gestionar Contenido.
- Internacionalizar.

- **Subsistema 2: Aplicación RACE Profile**

Constituido por los siguientes Casos de Uso:

- Gestionar Estilo de Aprendizaje.
- Identificar Tipo de Conexión.
- Gestionar Tipo de Conexión.
- Internacionalizar.

- **Subsistema 3: Servicio RACE Content**

Constituido por los siguientes Casos de Uso:

- Ensamblar Contenido.

- **Subsistema 4: Servicio RACE Agent**

Este módulo implementa una lógica de control que es utilizada o está incluida en algunos de los casos de uso de los tres módulos anteriores. También posee la capacidad de utilizar procesos definidos en LORS, LORSm y cronjob, como también usar programas accesibles mediante un shell (xsltproc, batch, nice, renice, etc.) y scripts en bash y perl que permiten controlar y simplificar las tareas de estilizado y manejo de esquemas IMS-SCORM. Por estas razones RACE Agent brinda soporte a los siguientes casos de uso:

- Publicar Contenido.
- Visualizar Contenido Personalizado.
- Exportar Contenido.

Descripción de los Casos de Uso esenciales del sistema:

Lineamientos para Generación de Contenidos Educativos y Rutas de Aprendizaje Personalizados para el Proyecto E-LANE en la Universidad del Cauca

Caso de Uso:	Gestionar Contenido
Actores:	Profesor (iniciador)
Tipo:	Primario
Descripción:	<ul style="list-style-type: none"> § El profesor accede a la herramienta con el fin de crear un nuevo contenido o modificar un contenido existente. § Crea el contenido o realiza los cambios necesarios si es un contenido existente. § Guarda el contenido. La información y los recursos son almacenados con una estructura acorde a docbook en el repositorio de contenidos de dotLRN/OpenACS.

Caso de Uso:	Ensamblar Contenido
Actores:	Profesor (iniciador)
Tipo:	Primario
Descripción:	<ul style="list-style-type: none"> § El profesor selecciona la opción de ensamblar contenido. § El contenido es accedido nuevamente (recuperado desde el repositorio) con el fin de crear los archivos docbook (.xml) de acuerdo a la estructura de ficheros establecida (books, chapters y sections). § Se recuperan los archivos correspondientes a los recursos y los archivos de las hojas de estilo XSL y CSS, las XSL son necesarias para poder presentar la información en el navegador del cliente (el estudiante) de una forma adecuada. Las CSS brindan la presentación final al contenido. Además el profesor podrá adicionar su propia hoja de estilo CSS para generar una presentación personalizada del contenido. § Una vez los archivos están creados, se genera el archivo: <i>imsmanifest.xml</i> que contiene una descripción de la estructura del contenido y su organización, utilizando el SCRIPT desarrollado por la Universidad Carlos III de Madrid.

Caso de Uso:	Publicar Contenido
Actores:	Profesor
Tipo:	Primario
Descripción:	<ul style="list-style-type: none"> § Todos los archivos son entregados directamente al paquete LORS que se encarga de publicarlos en el File Storage de la plataforma para quedar a disposición del estudiante.

Caso de Uso:	Exportar Contenido
Actores:	Profesor (iniciador)
Tipo:	Primario
Descripción:	<ul style="list-style-type: none"> § Posibilita al profesor exportar (descargar a su PC) a través de LORS un contenido almacenado en el repositorio en forma de paquete SCORM (archivo .ZIP), con el fin de que pueda ser reutilizado en cualquier otro LMS que soporte SCORM.

Caso de Uso:	Internacionalizar
Actores:	Usuario (profesor/administrador)
Tipo:	Secundario
Descripción:	<ul style="list-style-type: none"> § Este caso de uso posibilita la visualización de las interfaces de la aplicación Race editor en dos idiomas, Inglés y Español. Es posible dar soporte a cualquier otro idioma pero por el momento solo estarán disponibles los idiomas mencionados. § El usuario selecciona el idioma en el que desea visualizar la aplicación.

Lineamientos para Generación de Contenidos Educativos y Rutas de Aprendizaje Personalizados para el Proyecto E-LANE en la Universidad del Cauca

Caso de Uso:	Gestionar Estilo de Aprendizaje
Actores:	Estudiante (iniciador)
Tipo:	Primario
Descripción:	<ul style="list-style-type: none"> § El estudiante desarrolla el test de estilos de aprendizaje de David Kolb. § El sistema identifica su perfil de aprendizaje. § Si el estudiante no desea desarrollar el test, puede seleccionar su perfil de forma manual a través de una interfaz gráfica que le muestra los perfiles existentes junto con una descripción de cada uno de ellos.

Caso de Uso:	Identificar Tipo de Conexión
Actores:	Estudiante (iniciador)
Tipo:	Primario
Descripción:	<ul style="list-style-type: none"> § Cuando el estudiante accede a dotLRN, se inicializa una aplicación que detecta automáticamente el tipo de conexión a Internet que está utilizando. Se han definido dos tipos de conexión: <ul style="list-style-type: none"> ü Conexión de baja velocidad o banda estrecha (acceso conmutado). ü Conexión de alta velocidad o banda ancha (LAN, ADSL). § El sistema almacena el tipo de conexión detectada. § El tipo de conexión es desplegada al estudiante.

Caso de Uso:	Gestionar Tipo de Conexión
Actores:	Estudiante (Iniciador)
Tipo:	Primario
Descripción:	<ul style="list-style-type: none"> § El estudiante selecciona manualmente el tipo de conexión (baja o alta). Esta posibilidad es ofrecida en caso de que el estudiante desee acceder a los diferentes recursos del contenido asignados para cada tipo de conexión. § El sistema actualiza la información del tipo de conexión utilizada.

Caso de Uso:	Visualizar contenido Personalizado
Actores:	Estudiante (iniciador)
Tipo:	Primario
Descripción:	<ul style="list-style-type: none"> § Este es un caso de uso general que posibilita al estudiante recibir contenidos personalizados que hayan sido construidos por el profesor con la herramienta Race Editor y publicados en la plataforma.

4.3.3 Modelo de Análisis del Sistema

Paquetes y Clases de Análisis esenciales del Sistema

La siguiente figura muestra los paquetes de análisis de cada módulo en los que fue dividido el sistema.

Subsistema 1: RACE Editor

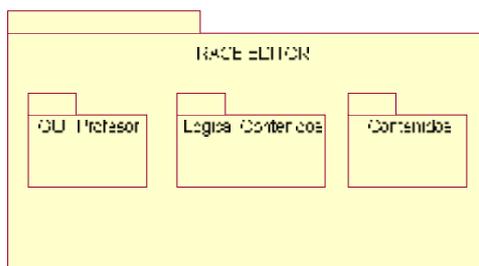


Figura 13 Diagrama de paquetes de análisis esenciales RACE Editor

Los paquetes *GUI_Profesor*, *Logica_Contenidos* y *Contenidos* incluyen las clases que implementan la funcionalidad del Caso de Uso Gestionar Contenido.

Posteriormente se definieron las clases de análisis asociadas a cada uno de los paquetes esenciales del módulo RACE Editor y a partir de estas se desarrollaron los diagramas de colaboración para los casos de uso esenciales del módulo (Para ver una descripción de cada clase y los Diagramas de Colaboración que muestran las relaciones y mensajes entre ellas, es necesario referirse al anexo A). A continuación se presentan algunos de los Diagramas de Clases de análisis del subsistema RACE Editor (Caso de Uso Gestionar Contenido), los Diagramas restantes se pueden encontrar en el anexo A.

Gestionar Contenido

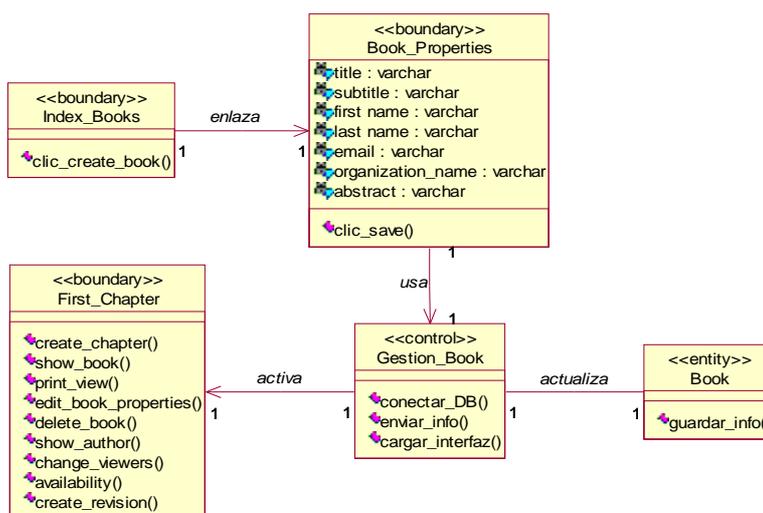


Figura 14 Diagrama de Clases de análisis Crear Book

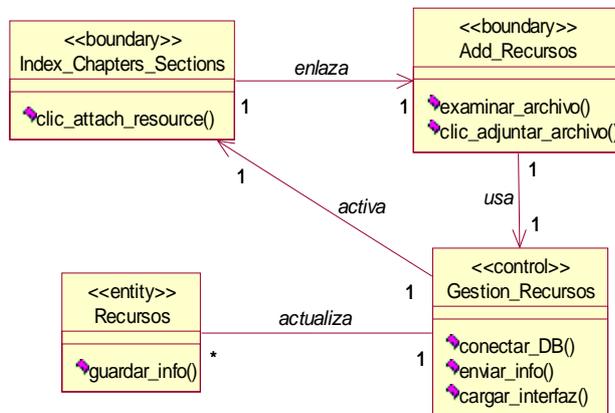


Figura 15 Diagrama de Clases de análisis Agregar Recurso

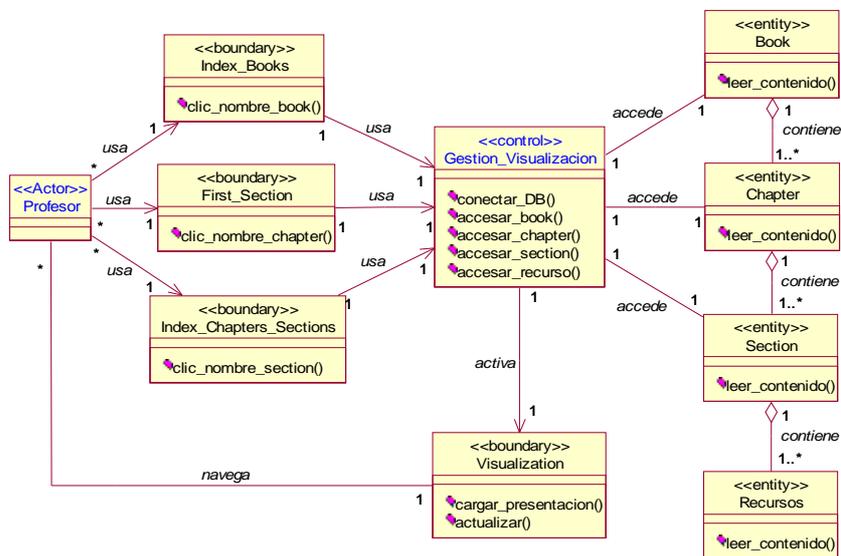


Figura 16 Diagrama de Clases de análisis Previsualizar Contenido

Subsistema 2: RACE Profile

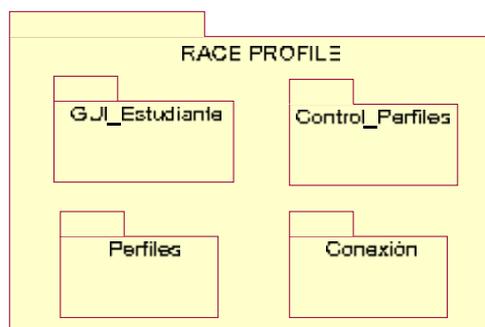


Figura 17 Diagrama de paquetes de análisis esenciales RACE Profile

Los paquetes *GUI_Estudiante*, *Control Perfiles* y *Perfiles* incluyen las clases que implementan la funcionalidad del Caso de Uso Gestionar Estilo de Aprendizaje y Gestionar Tipo de Conexión. El paquete *Conexión* incluye las clases que implementan la funcionalidad del Caso de Uso Identificar Tipo de conexión.

Posteriormente se definieron las clases de análisis asociadas a cada uno de los paquetes esenciales del módulo RACE Profile y a partir de estas se desarrollaron los diagramas de secuencia para los casos de uso esenciales del módulo (Para ver una descripción de cada clase y los Diagramas de Secuencia, es necesario referirse al anexo A). A continuación se presentan algunos de los diagramas de clases de análisis del subsistema RACE Profile (Casos de Uso Gestionar Estilo de Aprendizaje, Identificar Tipo de Conexión y Gestionar Tipo de Conexión), los diagramas restantes se pueden encontrar en el anexo A.

Gestionar Estilo de Aprendizaje

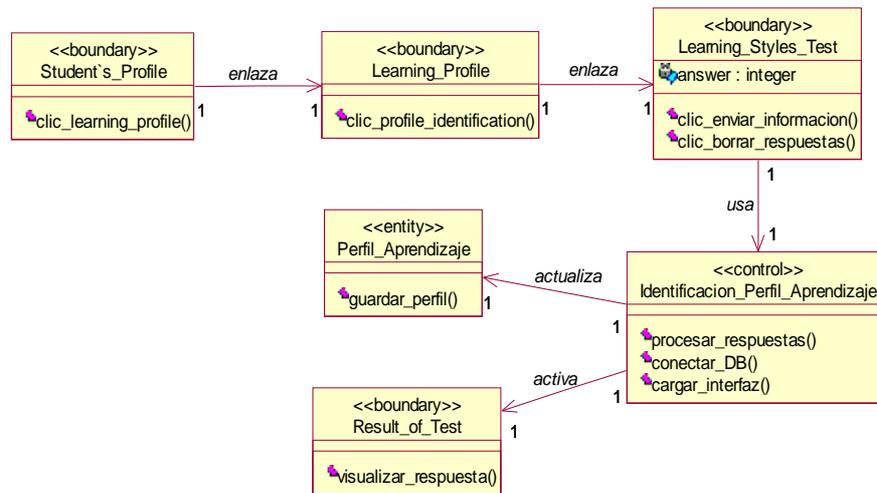


Figura 18 Diagrama de Clases de análisis Identificar Estilo de Aprendizaje

Identificar Tipo de Conexión

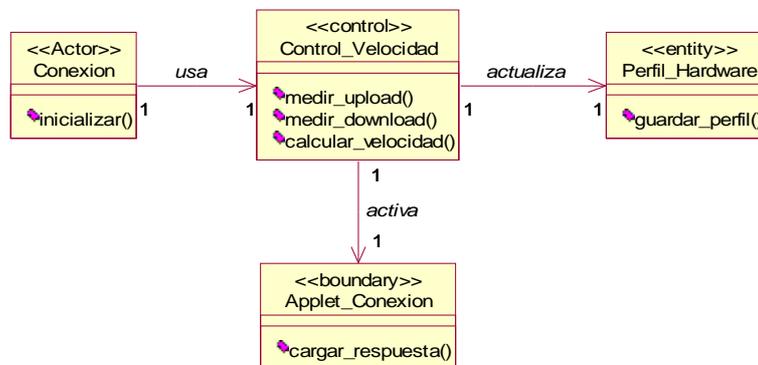


Figura 19 Clases de análisis Identificar Tipo de Conexión

Gestionar Tipo de Conexión

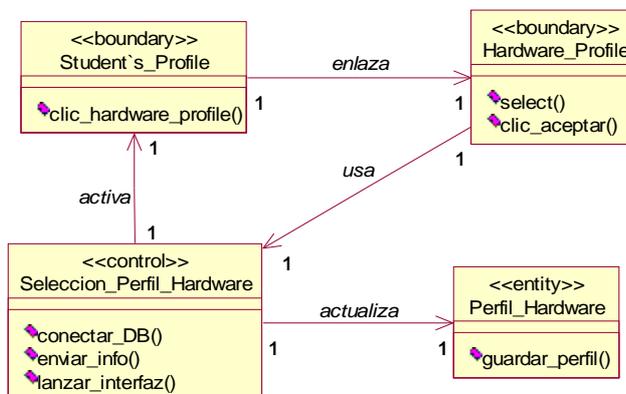


Figura 20 Clases de análisis Seleccionar Tipo de Conexión

Subsistema 3: RACE Content

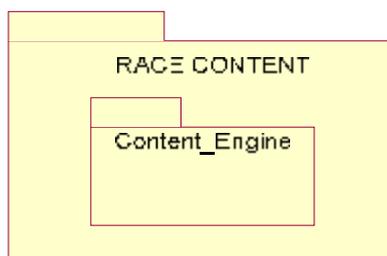


Figura 21 Diagrama de paquetes de análisis esenciales RACE Content

Los paquetes *GUI_Profesor*, *Content_Engine* y *Contenidos* incluyen las clases que implementan la funcionalidad del Caso de Uso Ensamblar Contenido.

Race Content suministra servicios de gestión de contenidos, con una serie de procedimientos para capturar, almacenar, solicitar, ensamblar, transformar y distribuir contenido (colocarlo en diferentes archivos de acuerdo a una estructura determinada como la que ofrece docbook).

Posteriormente se definieron las clases de análisis asociadas a los paquetes esenciales del módulo RACE Content (en este caso se cuenta con un solo paquete, sin embargo se deja abierta la posibilidad de incluir mayor funcionalidad mediante paquetes adicionales) y a partir

de estas se desarrollaron los diagramas de colaboración para los Casos de Uso esenciales del módulo (Para ver una descripción de cada clase y los Diagramas de Colaboración, es necesario referirse al anexo A). A continuación se presentan algunos de los diagramas de clases de análisis del subsistema RACE Content (Caso de Uso Ensamblar Contenido), los Diagramas restantes se pueden encontrar en el anexo A.

Ensamblar Contenido

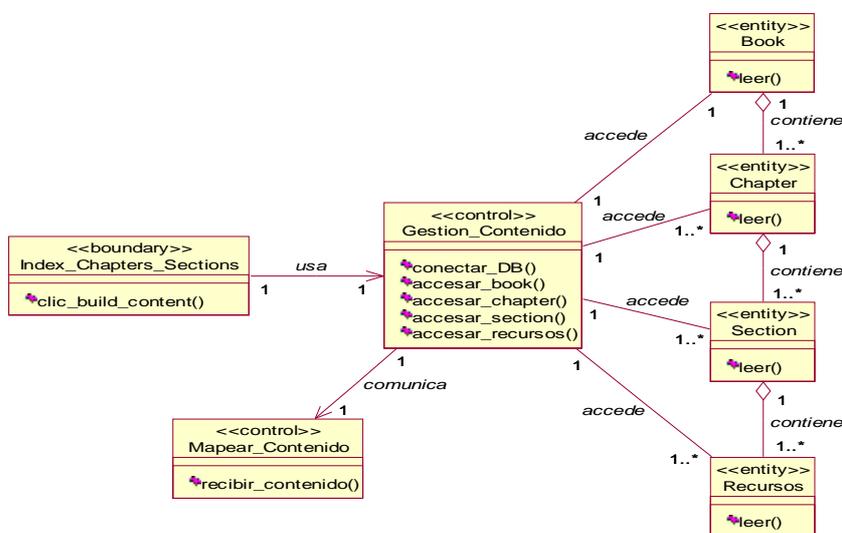


Figura 22 Diagrama de Clases de análisis Recuperar Contenido

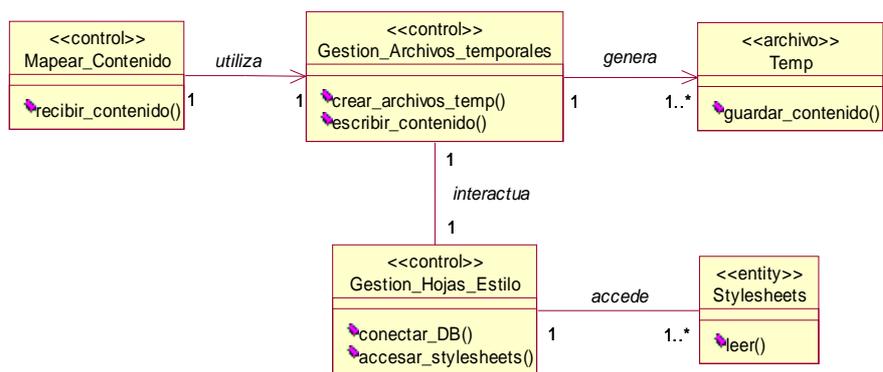


Figura 23 Diagrama de Clases de análisis Mapear Contenido y Agregar Estilo

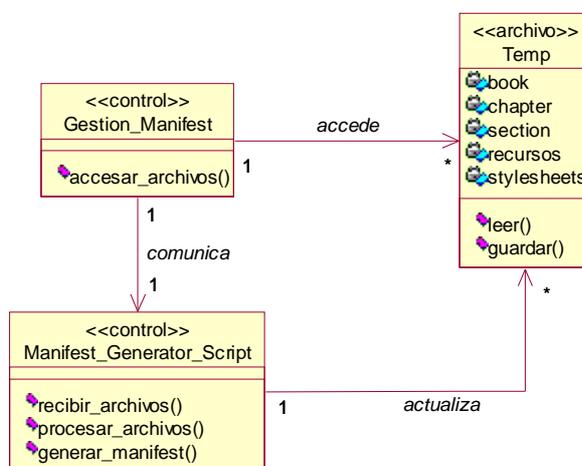


Figura 24 Diagrama de Clases de análisis Generar Manifest

Subsistema 4: RACE Agent

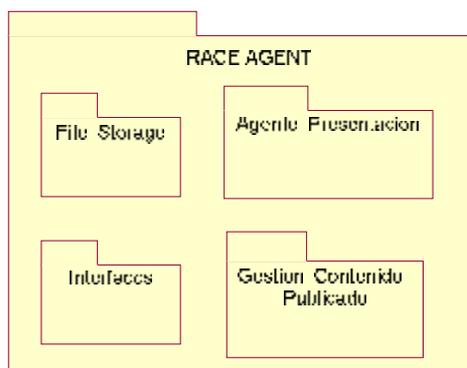


Figura 25 Diagrama de paquetes de análisis esenciales RACE Agent

Los paquetes *Content Engine* y *File Storage* incluyen las clases que implementan la funcionalidad del Caso de Uso Publicar Contenido. Los paquetes *Agente Presentación* e *Interfaces* incluyen las clases que implementan la funcionalidad del Caso de Uso Visualizar Contenido Personalizado. Los paquetes *GUI Profesor* y *Gestion Contenido Publicado* incluyen las clases que implementan la funcionalidad del Caso de Uso Exportar Contenido.

Posteriormente se definieron las clases de análisis asociadas a los paquetes esenciales del módulo RACE Agent y a partir de estas se desarrollaron los diagramas de colaboración para

los casos de uso esenciales del módulo (Para ver una descripción de cada clase y los Diagramas de Colaboración, es necesario referirse al anexo A). A continuación se presentan algunos de los diagramas de clases de análisis del subsistema RACE Agent (Casos de Uso Publicar Contenido, Visualizar Contenido Personalizado y Exportar Contenido), los diagramas restantes se pueden encontrar en el anexo A.

Publicar Contenido

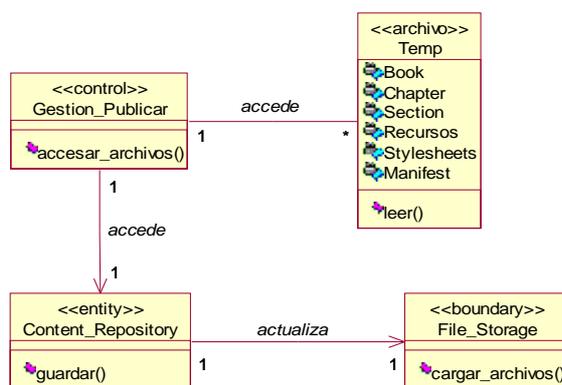


Figura 26 Diagrama de Clases de Análisis Publicar Contenido

Visualizar Contenido Personalizado

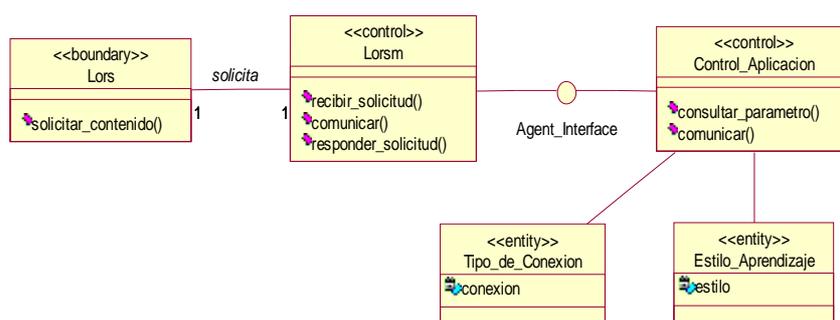


Figura 27 Diagrama de Clases de análisis Examinar Parámetros

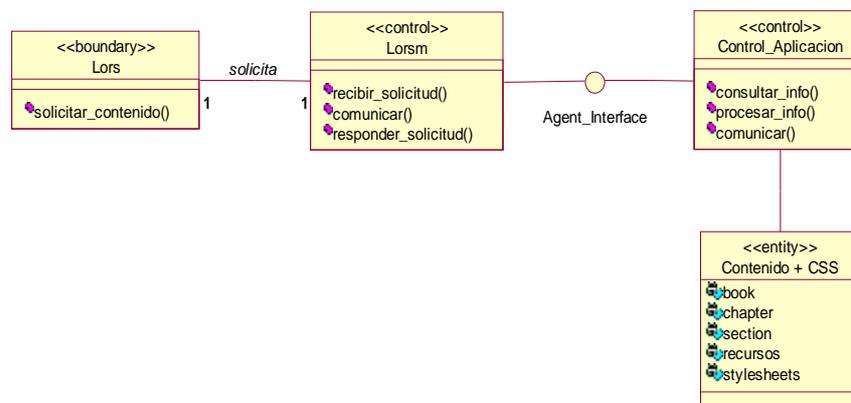


Figura 28 Diagrama de Clases de análisis Generar Presentación

Exportar Contenido

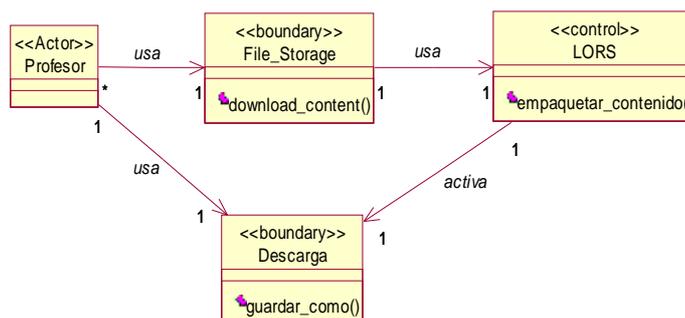


Figura 29 Diagrama de Clases de análisis Exportar Contenido

4.3.4 Modelo de Diseño del Sistema

A continuación se muestra una figura que contiene la estructura general del sistema a implementar, en ella se puede observar claramente cada uno de los subsistemas que conforman RACE y los elementos que los constituyen:

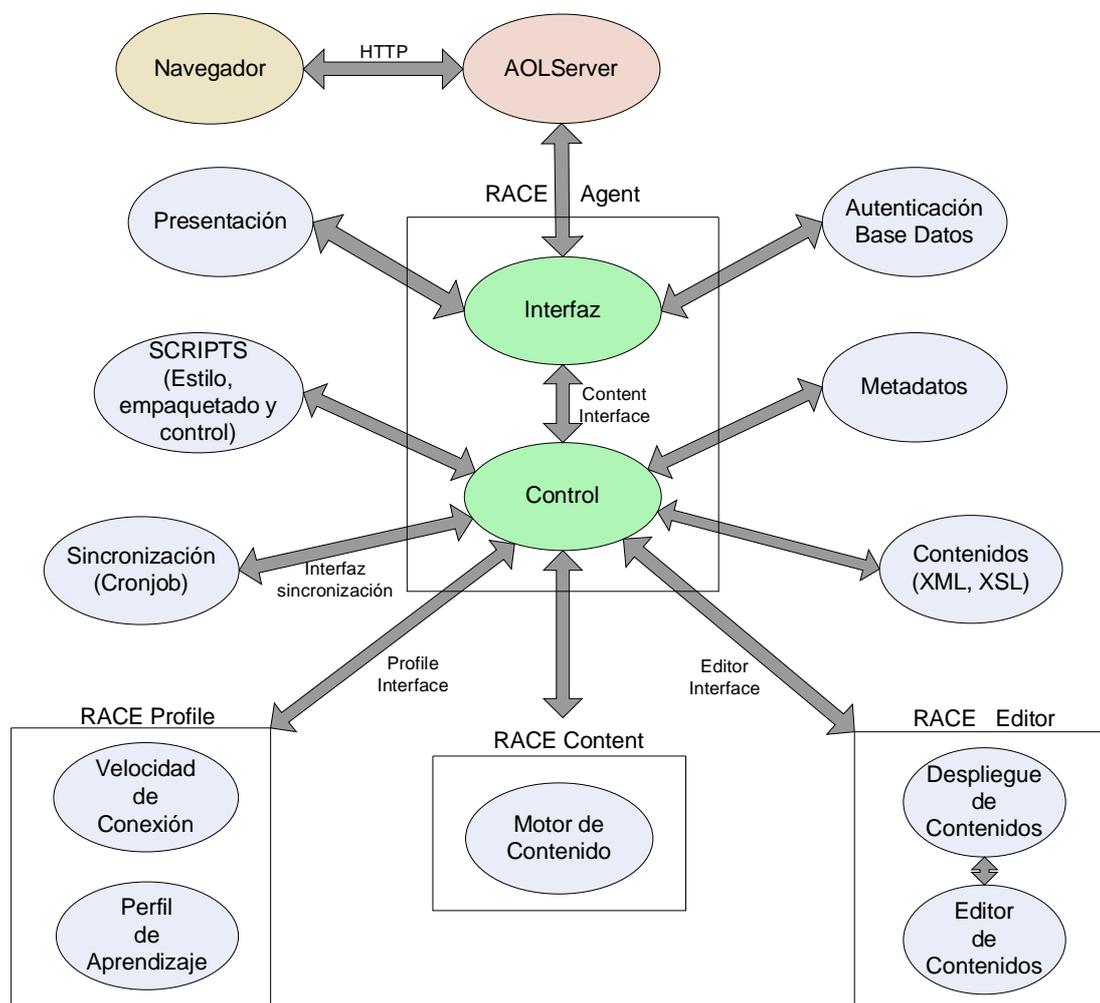


Figura 30 Estructura del sistema a implementar

- Race Agent:** RACE Agent está constituido por una parte de interfaz, cuya funcionalidad puede ser implementada en este caso por la aplicación LORSm, ya que permite que un estudiante pueda visualizar un contenido determinado almacenado mediante LORS en el file storage, de acuerdo a IMS Content Packaging, facilitando gran cantidad de métodos mediante el uso de interfaces, aunque esto implica limitarse a la implementación de una aplicación con poca realimentación por parte del estudiante. También se constituye de una lógica de control que se encarga la gestión de los otros subsistemas de RACE, y una lógica de comunicación con otras aplicaciones y programas externos a OpenACS.

- **Race Editor:** Contiene toda la lógica necesaria para la creación y edición de objetos de aprendizaje como también para la comunicación con RACE Agent cuando el usuario desea almacenar un objeto de aprendizaje en LORS.
- **Race Content:** Está formado por un motor de contenidos que suministra servicios de gestión y/o procesamiento de contenidos.
- **Race Profile:** Hay dos componentes en este módulo, el tipo de conexión del estudiante y su estilo de aprendizaje. Ambos aspectos son utilizados en el proceso de personalización de los contenidos.

Los demás elementos no son incluidos dentro de alguno de los subsistemas anteriores pero se destacan en la figura 29 debido al papel que cumplen dentro de todo el sistema. A continuación se describen estos componentes:

- **Scripts:** Son un conjunto de comandos para automatizar ciertas tareas de aplicación. Se utilizó dos scripts en bash, uno encargado de la transformación del contenido en html aplicando hojas de estilo XSL y el otro encargado del empaquetamiento de los contenidos HTML cumpliendo con la especificación content packaging de SCORM, finalmente otro script en perl que posibilita la utilización de texto enriquecido en los contenidos.
- **Cronjob:** Es un paquete de OpenACS que posibilita la sincronización de procesos. En este caso se encarga de sincronizar los Scripts con el sistema, pues se está teniendo en cuenta la carga del Servidor para llevar a cabo los diferentes procesos, de tal forma que sólo se ejecutan cuando el Servidor no está sobrecargado, lo que de otro modo podría implicar mucho tiempo de espera.
- **Autenticación:** Se utiliza el sistema de autenticación de OpenACS.
- **Metadatos:** Se pueden incluir los metadatos mediante el uso del script de empaquetamiento o también pueden ser agregados por el profesor gracias al uso de LORSm.

A partir de los Diagramas de Clases de análisis obtenidos en el modelo de análisis del sistema, se obtienen los diagramas de Clases de Diseño, a continuación se muestran

solamente algunas de estas clases, para una información mas completa de estos diagramas es necesario referirse al anexo A.

Subsistema 1: RACE Editor

Gestionar Contenido

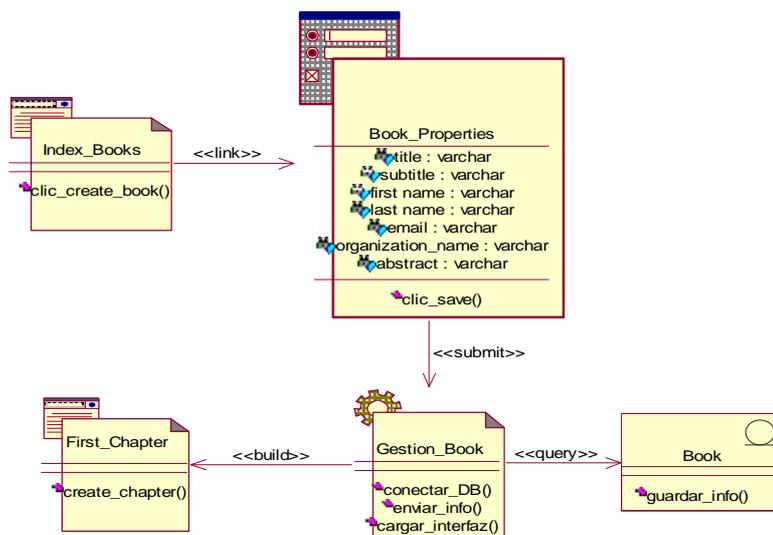


Figura 31 Diagrama de Clases de Diseño Crear Book

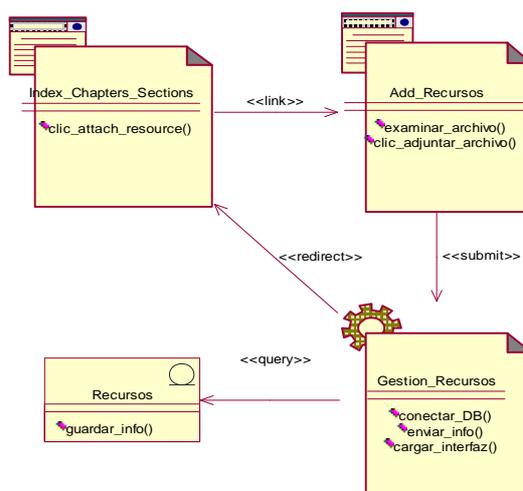


Figura 32 Diagrama de Clases de Diseño Adicionar Recurso

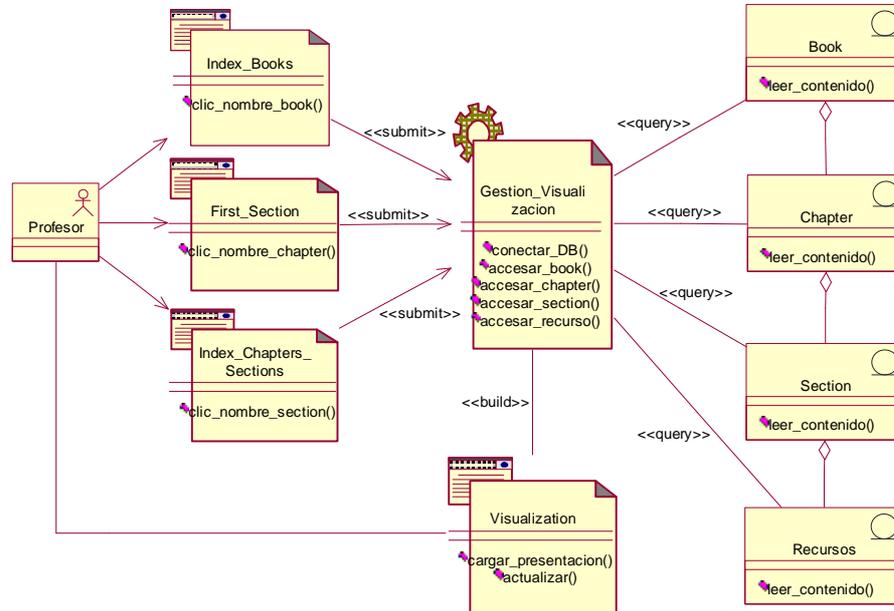


Figura 33 Diagrama de Clases de Diseño Previsualizar Contenido

Subsistema 2: RACE Profile

Gestionar Estilo de Aprendizaje

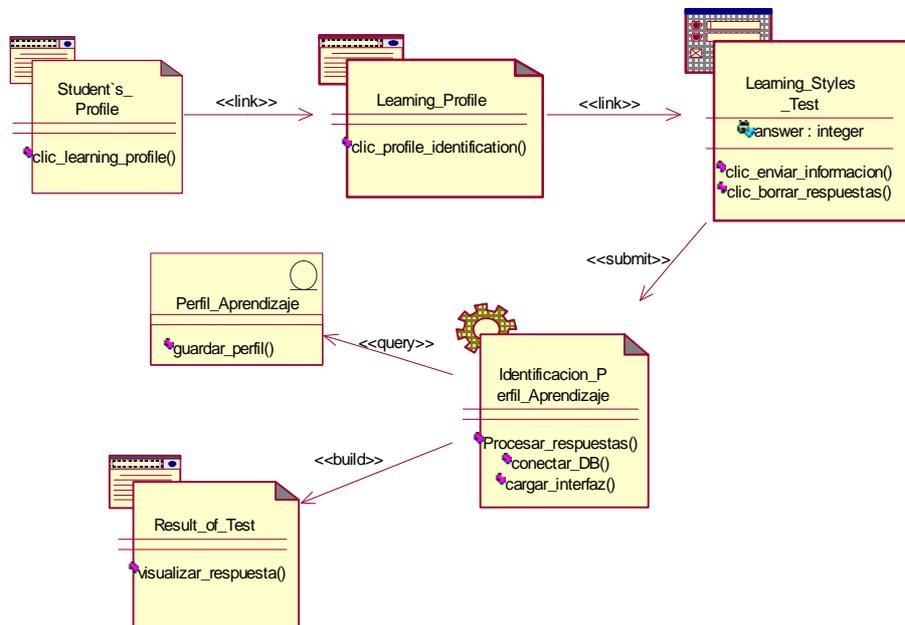


Figura 34 Diagrama de Clases de Diseño Identificar Estilo de Aprendizaje

Identificar Tipo de Conexión

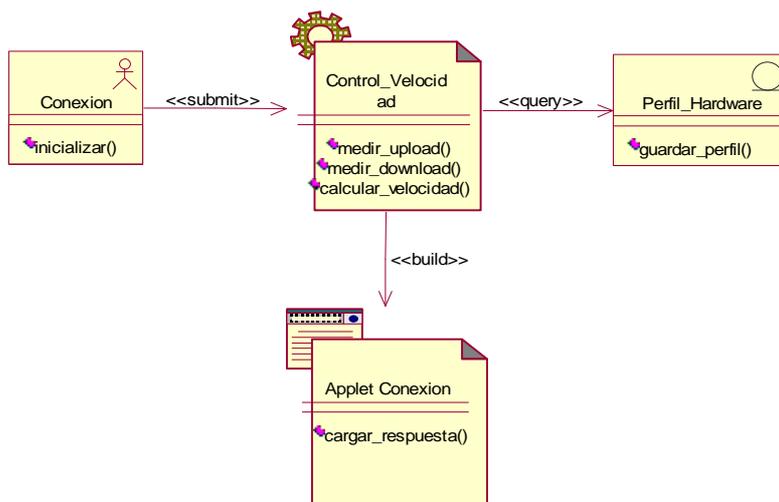


Figura 35 Diagrama de Clases de Diseño Identificar Tipo de Conexión

Gestionar Tipo de Conexión

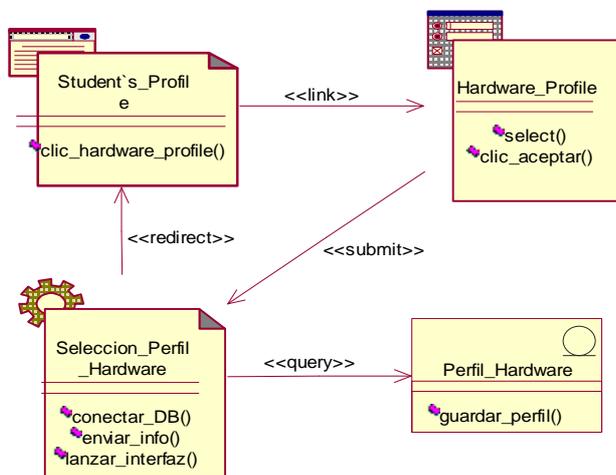


Figura 36 Clases de diseño caso de uso Seleccionar Tipo de Conexión

Subsistema 3: RACE Content

Ensamblar Contenido

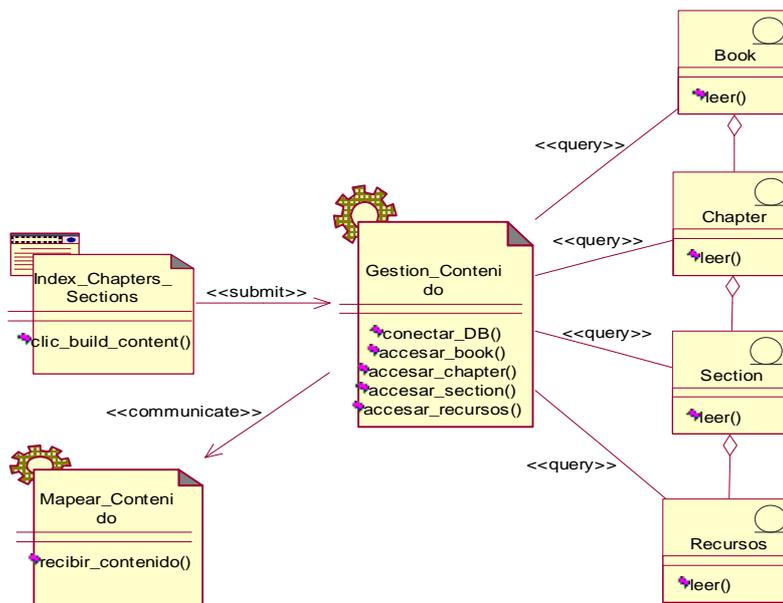


Figura 37 Diagrama de Clases de Diseño Recuperar Contenido

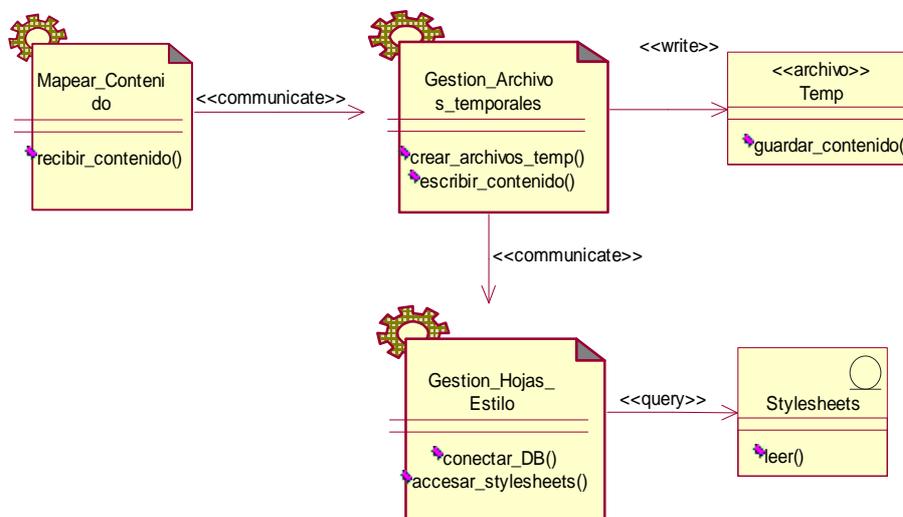


Figura 38 Diagrama de Clases de Diseño Mapear Contenido y Agregar Estilo

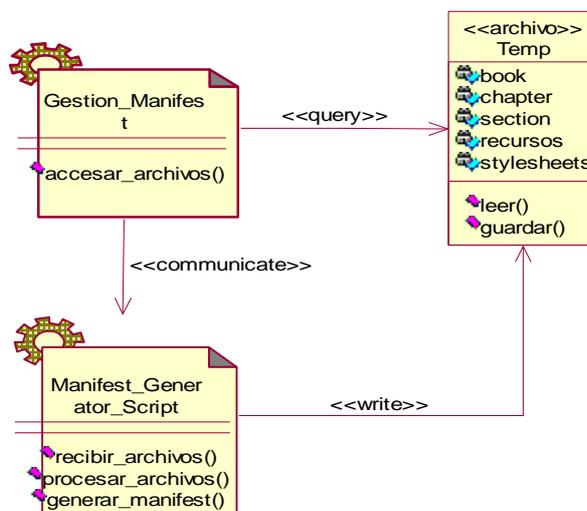


Figura 39 Diagrama de Clases de Diseño Generar Manifest

Subsistema 4: RACE Agent

Publicar Contenido

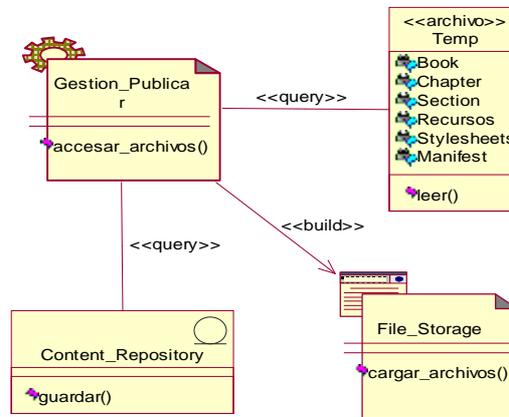


Figura 40 Diagrama de Clases de Diseño Publicar Contenido

Visualizar Contenido Personalizado

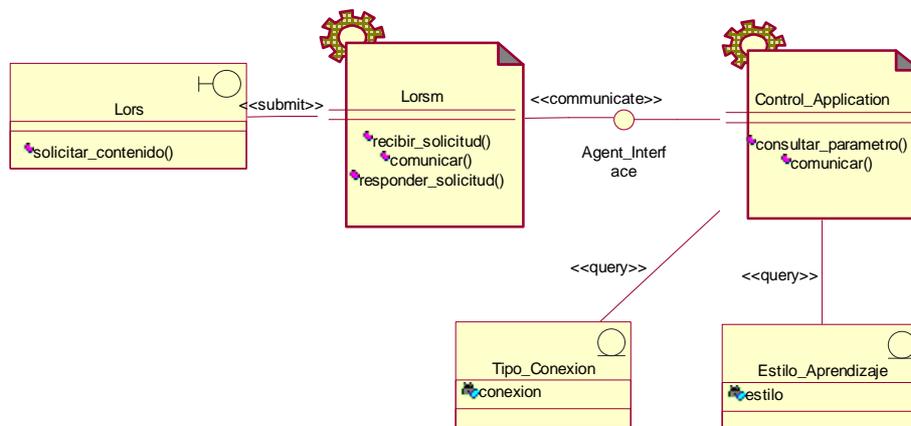


Figura 41 Diagrama de Clases de Diseño Examinar Parámetros

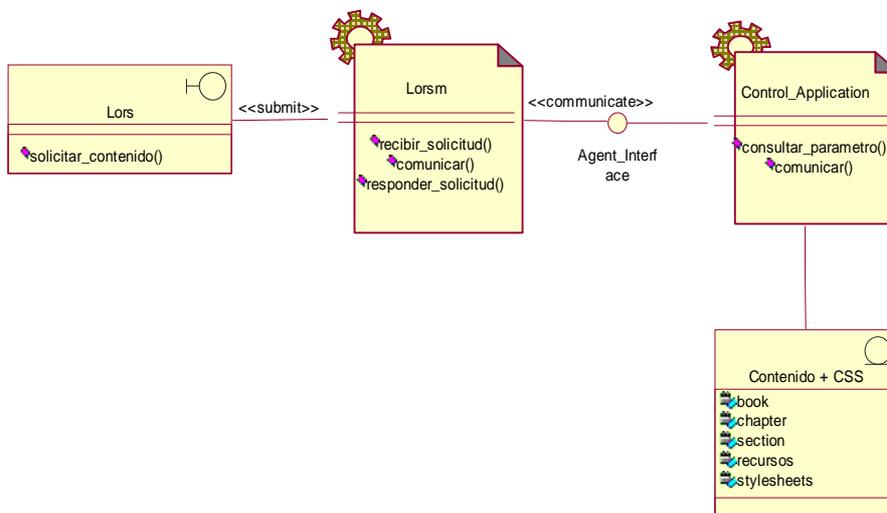


Figura 42 Diagrama de Clases de Diseño Generar Presentación

Exportar Contenido

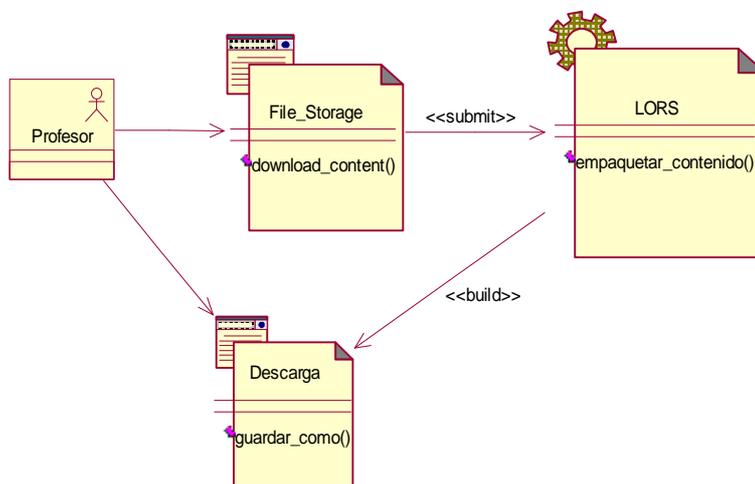


Figura 43 Diagrama de Clases de Diseño Exportar Contenido

4.3.5 Modelo de Implementación del Sistema

La figura 44 muestra el diagrama de Implementación del sistema.

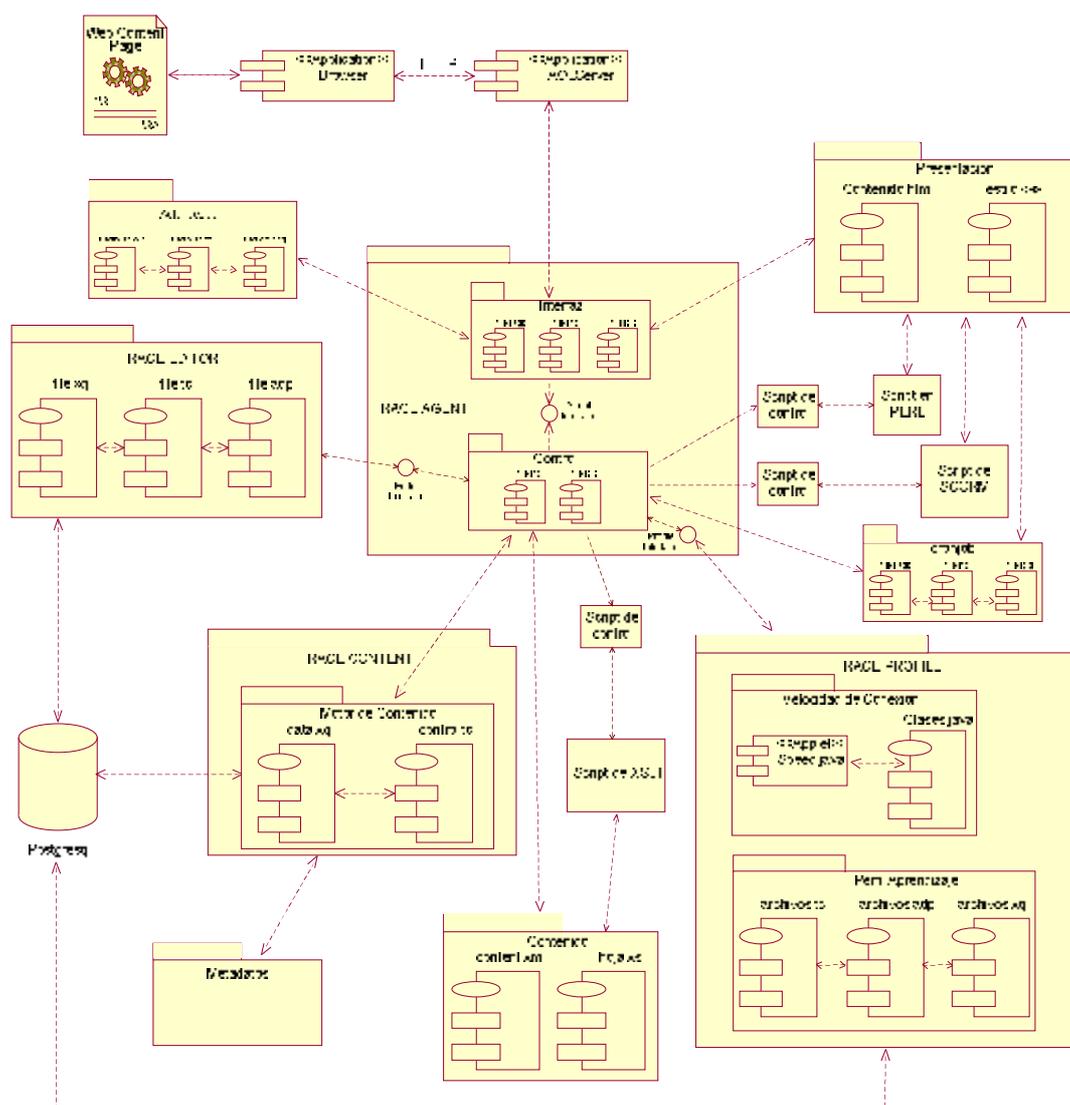


Figura 44 Diagrama de Implementación del Sistema

En la figura 44 se puede observar la estructura, a grosso modo, del código del sistema RACE, además componentes y scripts que ayudan a su funcionamiento, con los subsistemas correspondientes como son el editor, el motor de contenido (content), el subsistema de

En la figura 45 se muestran las relaciones existentes entre los distintos componentes hardware y software del sistema RACE y su entorno, dicho de otro modo, la configuración de los elementos de procesamiento del sistema final en tiempo de ejecución y los componentes software. En la figura se observa tres nodos que son un servidor Web, un servidor de base de datos y un navegador Web en el lado del PC del estudiante, la configuración de los componentes de RACE sobre OpenACS es mostrada en el PC sobre el cual se encuentra corriendo el servidor Web AolServer.

5 CASO DE ESTUDIO

5.1 CONSTRUCCIÓN DE UN PROTOTIPO DE CONTENIDO EDUCATIVO DIRIGIDO A ESTUDIANTES CON DIFERENTES PERFILES

Para la validación de la guía metodológica generada en el capítulo 3 y las herramientas que constituyen el sistema RACE, se construyó un prototipo de contenido educativo que incluye generalidades sobre el Lenguaje de Modelado Unificado UML. El contenido fue desarrollado utilizando RACE Editor con el fin de poner a prueba su funcionalidad. No se utilizó ninguna de las aplicaciones útiles en la configuración del entorno virtual de aprendizaje mencionadas en el capítulo 4. A continuación se muestra la aplicación de la guía metodológica al prototipo desarrollado.

1. Inicialmente se definió la temática que se pretende dar a conocer a los estudiantes con el fin de establecer los objetivos y delimitar los alcances del contenido educativo. Posteriormente, se realizó una búsqueda de material educativo existente referente al tema a desarrollar, disponiendo de una buena fuente de información proveniente del contenido de la materia Ambientes de Desarrollo, cursos sobre UML y documentos publicados al interior del Departamento de Telemática. Se encontró información teórica y recursos como imágenes, animaciones, audio y video que fueron analizados para seleccionar la información más adecuada. Los objetivos definidos para el contenido educativo fueron:

- Conocer los aspectos fundamentales sobre UML, como lenguaje de notación gráfico para el modelado de sistemas software.
 - Dominar el vocabulario, las reglas y las construcciones específicas de UML.
 - Comprender cómo utilizar UML para resolver varios problemas de modelado comunes.
 - Informarse sobre las diferentes fases que constituyen un proceso de desarrollo con UML.
2. Antes de comenzar a desarrollar el contenido se definió su estructura de forma jerárquica realizando una fragmentación lógica en capítulos y secciones. El tema fue dividido en dos capítulos de la siguiente forma:
- Capítulo 1: Las Vistas UML. Contiene una descripción de cada una de las cinco vistas que plantea UML para el modelado de la arquitectura de un sistema software y los diagramas que posibilitan su descripción. Está constituido por las siguientes secciones:
 - ü Vista de Casos de Uso, vista lógica, de componentes, implantación y concurrencia.
 - ü Diagrama de Casos de Uso.
 - ü Diagrama de Clases.
 - ü Diagrama de Paquetes.
 - ü Diagramas de Colaboración y Secuencia.
 - ü Diagramas de estados y actividad.
 - ü Diagramas de Componentes e implantación.
 - Capítulo 2: Mecanismos de extensión de UML. Contiene una descripción de algunos elementos sintácticos que permiten ampliar la capacidad descriptiva que ofrece UML. Gracias a estos mecanismos de extensión se puede expresar una gran variedad de conceptos utilizados en diferentes dominios de una aplicación. Este capítulo está constituido por las siguientes secciones:

- ü Valores etiquetados.
- ü Restricciones.
- ü Estereotipos.

La figura 46 muestra la división del contenido del contenido en capítulos, conformados por sus respectivas secciones.



Figura 46 Estructura del Contenido Desarrollado

3. En cuanto a la duración del contenido, se desarrolló una sola sesión o clase debido a que es un prototipo de contenido educativo para validar el sistema implementado en el presente proyecto. El tiempo estimado para la apropiación del contenido de dicha sesión por parte de un estudiante es aproximadamente de 2 horas y el estudiante tiene la posibilidad de acceder en cualquier momento y por el tiempo que considere necesario hasta su finalización. Además si el profesor desea dar algún tipo de seguimiento al

estudiante, puede utilizar algunas aplicaciones de dotLRN como las nombradas en el capítulo anterior (attendance, assessment, survey, etc.).

4. Dado que RACE Editor utiliza la presentación por defecto propuesta por el proyecto E-LANE [76], se empleó la distribución espacial sugerida por dicha presentación. La figura 47 muestra la distribución de los elementos que constituyen una página de contenido en pantalla.

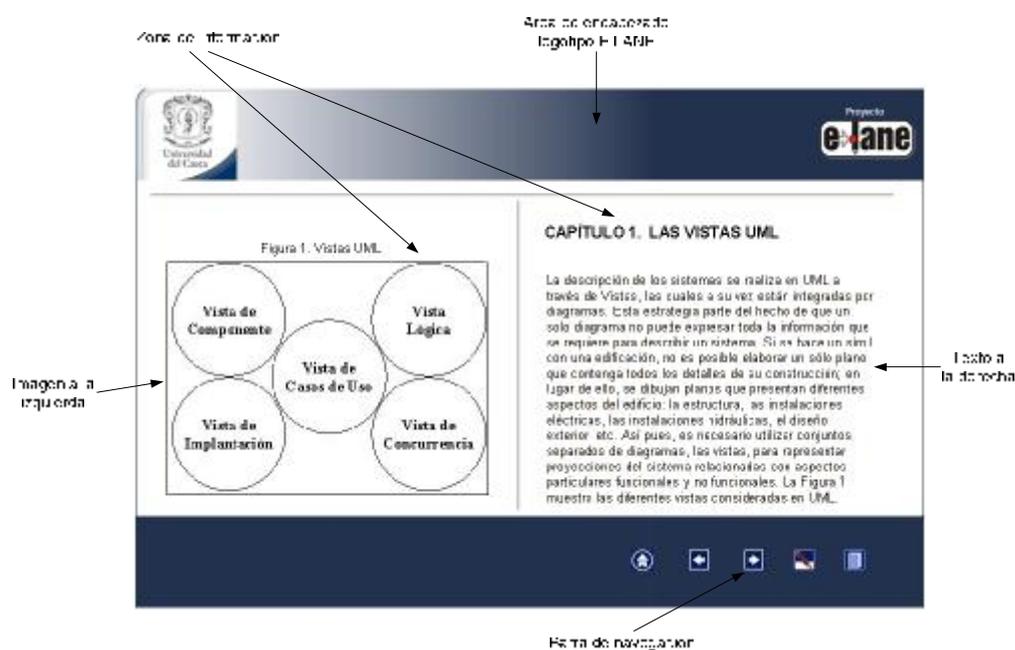


Figura 47 Distribución espacial para el desarrollo del contenido

5. Como se muestra en la figura 48, los contenidos se enfocaron hacia los estilos de aprendizaje convergente y asimilador (dos de los estilos definidos en el modelo de Kolb) considerando velocidades de conexión baja y alta. Para cada estilo se desarrolló una actividad de aprendizaje de acuerdo a los aspectos que lo caracterizan (la descripción de las características de estos estilos se puede encontrar en el anexo A.). Los recursos a incluir se definieron según la velocidad de conexión, teniendo en cuenta la información suministrada en el capítulo 3 de este documento, es decir, usar recursos multimedia livianos para el caso de velocidad de conexión baja. En el numeral 7 se especifican las

tareas que se asignaron a los perfiles mencionados anteriormente y en el numeral 8 se visualiza el contenido de dichas tareas.

Crear Book

My Space My Calendar My Files Control Panel

Ingrese un título para el book

*** Título:**

Agregue un subtítulo si lo desea

Subtítulo:

autor Información personal sobre el creador del contenido

Nombre:

Apellido:

email:

Nombre de la institución a la que pertenece el profesor:

Organización:

Aquí puede introducir un pequeño resumen sobre el contenido del book

Resumen:

Estilo de aprendizaje: Seleccione el perfil para el cual va dirigido este contenido:

Perfil predeterminado
Convergente
Asimilador
Divergente
Acomodador

Velocidad de conexión: Seleccione el tipo de conectividad:

predeterminada
Alta
Baja

Guardar

Figura 48 Selección del estilo de aprendizaje y el tipo de conectividad

6. A continuación se muestran los objetivos definidos para algunas de las unidades didácticas del curso:

§ Diagrama de Casos de Uso:

- ü Comprender los conceptos de caso de uso, actor, asociaciones y relaciones de inclusión y extensión.
- ü Desarrollar habilidades para el modelado del comportamiento de un sistema.

§ Diagrama de Clases:

- ü Comprender los conceptos de clase, atributos y operaciones.

- ü Diferenciar con claridad las relaciones de dependencia, generalización y asociación.
- ü Empezar a utilizar el concepto de herencia.

§ Diagrama de Colaboración y secuencia:

- ü Aprender a implementar diagramas de interacción (colaboración y secuencia).
- ü Comprender la diferencia existente entre lo diagramas de secuencia y colaboración.

The image shows a screenshot of a digital learning interface. At the top left is the logo of the Universidad del Cauca. At the top right is the logo for 'Proyecto e-lane'. The main content area is titled 'Sección 1. Diagrama de Casos de Uso'. Below the title is a 'Tabla de Contenido' with the following links: [objetivos](#), [conceptos de caso de uso](#), [concepto de actor](#), [concepto de asociación](#), and [relaciones de inclusión y extensión](#). Below the table of contents is the 'Objetivos' section, which contains two bullet points: '* Comprender los conceptos de caso de uso, asociaciones y relaciones de inclusión y extensión.' and '* Desarrollar habilidades para el modelado del comportamiento de un sistema.' At the bottom of the interface is a dark blue navigation bar with icons for home, back, forward, search, and a document icon.

Figura 49 Ejemplo de los objetivos definidos para una unidad didáctica

7. Ahora se muestran algunos aspectos en el desarrollo de contenidos con soporte a los estilos de aprendizaje asimilador y convergente, teniendo en cuenta también la velocidad de conexión.

Para el perfil asimilador + conectividad baja. Se incluyó una lectura sobre el desarrollo de un Diagrama de Casos de Uso de un problema en particular, además se incluyó en la

lectura ejemplos completos de casos similares. Por la condición de conectividad baja se decidió incluir en el contenido información textual, videos e imágenes livianas.

Para el perfil asimilador + conectividad alta. La misma actividad anterior pero incluyendo videos de alta calidad.

Para el perfil convergente + conectividad baja. Como al estudiante convergente le gusta aplicar de forma práctica la teoría aprendida, se incluyó un ejercicio a través del cual tiene la posibilidad de implementar el Diagrama de Casos de Uso que se le muestra en el contenido mediante la utilización de alguna herramienta software para modelado como Umbrella, Rational Rose, etc. Para ello se le muestra una animación con calidad baja sobre el uso de dichas herramientas software.

Para el perfil convergente + conectividad alta. Se incluyó el mismo ejercicio del perfil anterior apoyado en un video de calidad alta instructivo que muestra cómo utilizar la herramienta Rational Rose para el desarrollo de un Diagrama de Casos de Uso. En él se muestra paso a paso cómo crear el diagrama y los diferentes aspectos que se debe tener en cuenta. Además, para este tipo de perfil, en una actividad posterior, se incluyó enlaces hacia una herramienta de modelado a nivel de Web con la cual se pueden realizar diagramas de clases como el mostrado en la figura 50.

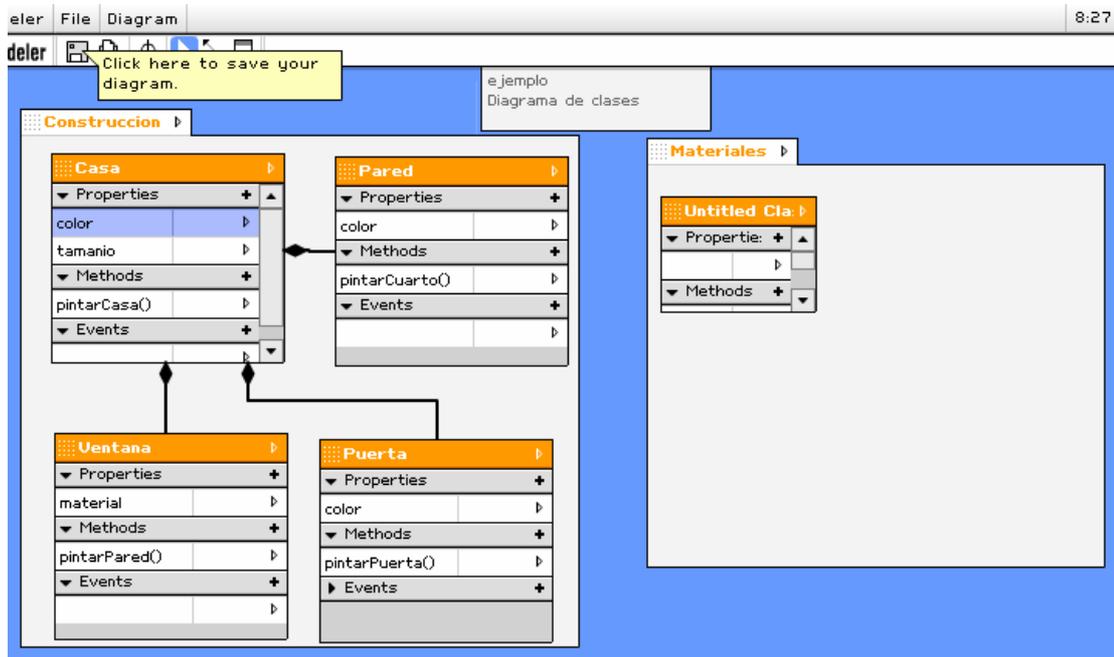


Figura 50 Herramienta de modelado a nivel Web: gmodeler [95]

8. Para crear el contenido de cada actividad de aprendizaje se incluyó información existente, tomada de [96] y [97]. Las siguientes figuras muestran los contenidos creados para cada perfil de estudiante.

§ Contenido desarrollado para un estudiante asimilador con conectividad baja.

The screenshot displays a web interface for 'Proyecto e-lane' from the Universidad del Cauca. It features a UML Use Case diagram for a library management system. The diagram shows four actors: Lector (Reader), Usuario (User), Director, and SI_Administrativo (System Administrator). The Lector actor is associated with 'Hacer consulta' (Make inquiry) and 'Hacer reserva' (Make reservation). The Usuario actor is associated with 'Control acceso' (Access control). The Director actor is associated with 'Gestionar monitores' (Manage monitors) and 'Registrar nuevo lector' (Register new reader). The SI_Administrativo actor is associated with 'Consultar multas' (Check fines). The Monitor actor is associated with 'Borrar reserva' (Delete reservation) and 'Prestar item' (Lend item). The text box on the right, titled 'Lectura ejemplo' (Example reading), contains the text: 'GESTIÓN DE UNA BIBLIOTECA. La figura muestra un diagrama de casos de uso que describe parcialmente un sistema para la gestión de una biblioteca. El sistema tiene cuatro actores: Lector, Monitor, Director y Administrativo. Los casos de uso son: Hacer Consulta, permite buscar un recurso en la biblioteca con opción de reserva. Hacer Reserva, asignar a un lector un libro para que pueda retirarlo más tarde. Borrar Reserva, eliminar la asignación de un libro a un'. Below the text box is a link: 'sistema-biblioteca'.

Figura 51 Contenido para el estudiante asimilador + conectividad baja

Dado que un estudiante con estilo de aprendizaje asimilador se siente más a gusto con contenidos teóricos con los cuales pueda contextualizar y reflexionar, el contenido para este estilo de aprendizaje va orientado hacia la presentación de conceptos y desarrollos de ejercicios claros, así como hacia la formulación de preguntas de las cuales pueda surgir debate, entre otros aspectos mostrados para este perfil en el Anexo B.

§ Contenido desarrollado para un estudiante asimilador con conectividad alta.

La interfaz para este perfil es la misma que la mostrada en la figura 51 con la diferencia que la calidad del video en este caso es superior ya que la conectividad soporta este recurso.

§ Contenido desarrollado para un estudiante convergente con conectividad baja.

Universidad del Cauca

Proyecto e-lane

Diagrama de Casos de Uso

EJERCICIO: IMPLEMENTACIÓN DEL DIAGRAMA DE CASOS DE USO DEL SISTEMA DE GESTIÓN DE LA BIBLIOTECA

Mediante el uso de alguna herramienta software para modelado como Umbrella o Rational Rose, implemente el diagrama de casos de uso que se observa en la figura de la izquierda, aplicando los conceptos de relación: <<extend>> e <<include>>, y los conceptos adicionales de agregación, generalización, etc. vistos anteriormente.

A continuación se presenta una animación que ofrece una guía básica sobre el manejo de la herramienta Rational Rose. Esta guía le puede ser de gran utilidad para el desarrollo del ejercicio anterior:

[Guía](#)

Figura 52 Contenido para el estudiante convergente + conectividad baja

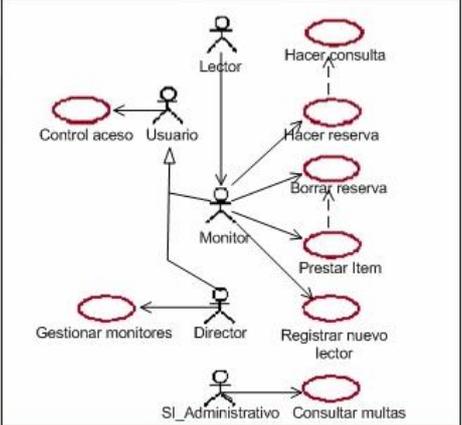
En el modelo de Kolb, a un estudiante con estilo de aprendizaje convergente le gusta aplicar de manera práctica las ideas, entre otros muchos aspectos, el contenido para este estudiante se crea de tal modo que se le de libertad justamente para poner en práctica lo aprendido.

§ Contenido desarrollado para un estudiante convergente con conectividad alta.

En este caso, al igual que para el estilo de aprendizaje asimilador con velocidad de conexión alta, solo varía la calidad y/o tamaño (o necesidad de velocidad de conexión suficiente) de los recursos utilizados.



Diagrama de Casos de Uso



EJERCICIO: IMPLEMENTACIÓN DEL DIAGRAMA DE CASOS DE USO DEL SISTEMA DE GESTIÓN DE LA BIBLIOTECA

Mediante el uso de alguna herramienta software para modelado como Umbrella o Rational Rose, implemente el diagrama de casos de uso que se observa en la figura de la izquierda, aplicando los conceptos de relación: <<extend>> e <<include>>, y los conceptos adicionales de agregación, generalización, etc. vistos anteriormente.

En el siguiente video se muestra el procedimiento que usted debe seguir para crear el diagrama y los diferentes aspectos que debería considerar.
[video-rational-rose](#)

En el siguiente enlace encontrará una herramienta Web que le podría ser útil en el desarrollo de algunos diagramas de UML. Su nombre es "g-modeler".
[g-modeler](#)



Figura 53 Contenido para el estudiante convergente + conectividad alta

9. información agregada al contenido educativo:

Título del contenido: Lenguaje Unificado para la Construcción de Modelos.

Autor: Ing. Alvaro Rendón.

Abstract: Curso sobre UML, de prueba para el editor RACE.

Capítulo 1:

Título: Vistas UML

Objetivos: Metas esperadas al finalizar este capítulo

Capítulo 2:

Título: Mecanismos de Extensión de UML

Objetivos: Metas esperadas al finalizar este capítulo

Además se agregaron metadatos al contenido educativo mediante LORS, gracias al soporte que ofrece este paquete para la inserción de metainformación.

10. Gracias a RACE Editor se brindó soporte para realizar revisiones y adicionar comentarios al contenido.



Figura 54 Revisión adicionada al contenido

11. Una vez el contenido fue culminado, se publicó en el sistema repositorio de objetos de aprendizaje de la plataforma dotLRN/OpenACS.

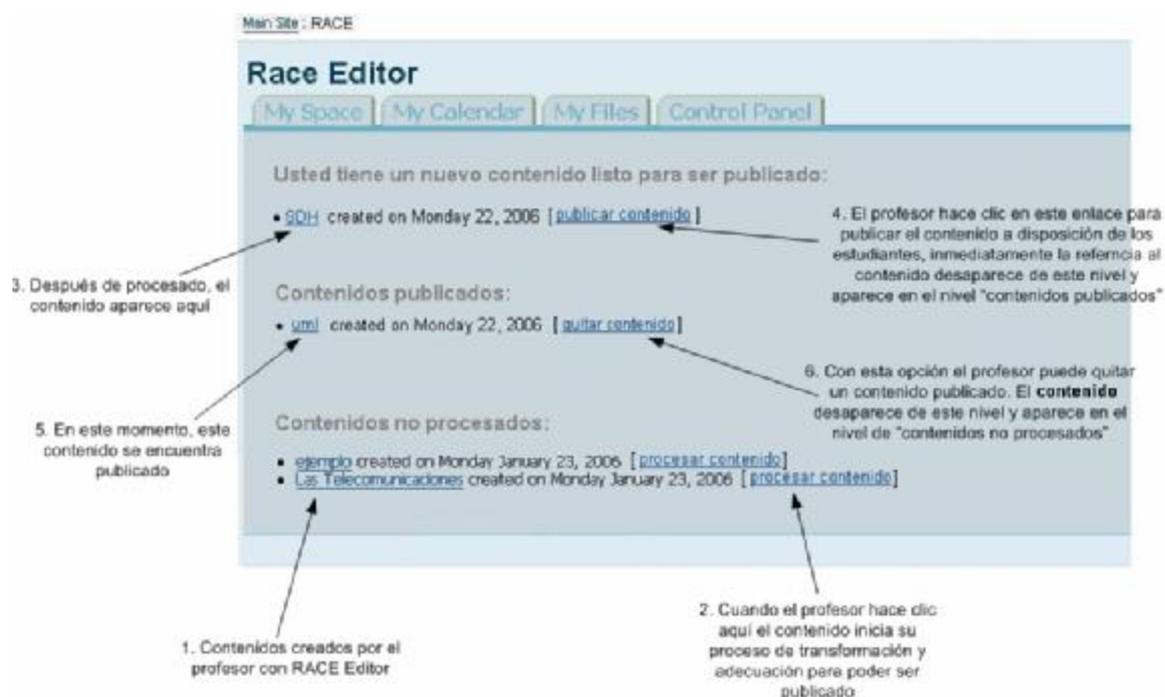


Figura 55 Publicación del contenido en LORS

12. Para la evaluación de los contenidos creados se recomienda medir su calidad considerando los siguientes aspectos:

- La facilidad en la comprensión del contenido por parte de los estudiantes.
- Utilización adecuada de distintos materiales de aprendizaje de acuerdo a estilos de aprendizaje y velocidad de conexión para la generación del contenido por parte del profesor.
- Grado de personalización del contenido de acuerdo al perfil del estudiante.
- Grado de satisfacción del estudiante con respecto al contenido.

Como trabajo futuro se debe considerar la evaluación del contenido educativo por parte de los estudiantes para medir el nivel de aceptación del contenido y la valoración que ellos otorguen al mismo, todo con soporte por parte de RACE, es decir todo en un entorno Web.

5.2 ASPECTOS TECNOLÓGICOS CONSIDERADOS

Teniendo en cuenta las recomendaciones tecnológicas formuladas en la metodología propuesta en el capítulo 3 se consideraron los siguientes aspectos:

- § Se ofreció contenidos adaptados a la capacidad de la conexión utilizada por el estudiante, para conectividad baja se empleó imágenes, videos y audio en formatos livianos y texto, para conectividad alta se utilizó archivos de imágenes, video y audio de calidad alta.
- § Se manejó dos calidades de un mismo recurso, en el caso de recursos de tipo flash para utilizarse en cada tipo de conectividad.

5.3 ASPECTOS DE USABILIDAD CONSIDERADOS

El campo de usabilidad es muy amplio, sin embargo se propone tenerlo en cuenta a la hora de desarrollar contenidos educativos dada su importancia actualmente. Teniendo en cuenta algunas recomendaciones tomadas del gran campo de la usabilidad propuestas en el capítulo 3 se consideraron los siguientes aspectos:

- § El tema se desarrolló incluyendo una sola idea en cada párrafo evitando saturación del contenido que pudiera generar indisposición al estudiante durante el proceso de lectura.
- § Se utilizó un lenguaje objetivo y simple con un estilo de redacción apropiado para los documentos Web posibilitando una lectura más rápida. Al principio del tema se colocó la idea principal del mismo ofreciendo una idea global de lo que se va a leer.
- § Como lo muestran las figuras incluidas en el numeral 8, se tuvo en cuenta (gracias a RACE Editor) la navegabilidad sobre el contenido al utilizar una plantilla con una barra de navegación adecuada en la parte inferior de la pantalla. Por otra parte, el contenido educativo cumple con algunos criterios de accesibilidad en cuanto a tipo de navegador, por eso los contenidos se probaron en Explorer, Firefox, Konqueror, y adicionalmente lynx; en los 3 primeros el contenido se desplegó sin mayor problema, en el último, dado que es un navegador orientado a texto se tuvo en cuenta aspectos como el uso de texto adecuado (atributo Alt) en el lugar donde existían imágenes.

- § Se utilizó hipervínculos al interior del contenido (uso de texto enriquecido) que posibilitan el enlace a diferentes partes del contenido de una forma inmediata y a otras fuentes de información que pueden complementar el tema.
- § No sobra aclarar que RACE Editor, de manera predefinida, cumple con lineamientos como distribución espacial adecuada y uniforme de los contenidos, navegabilidad, entre otros.

Finalmente se tuvo en cuenta el uso de herramientas de escritorio (último lineamiento del capítulo 3) para edición de imágenes, video, etc. De esta manera concluye este capítulo que busca validar los lineamientos planteados en el capítulo tres de esta monografía, a la vez que valida el soporte que para gran parte de dichos lineamientos ofrece RACE. Los resultados fueron satisfactorios, tanto a nivel de los objetivos buscados como a nivel de satisfacción de los desarrolladores, dejando abiertas gran cantidad de posibilidades de extensión tanto de los lineamientos como de RACE en la búsqueda de la creación de contenidos de alta calidad y bajo costo como pretende el proyecto E-LANE al interior de la Universidad del Cauca.

CONCLUSIONES, RECOMENDACIONES Y TRABAJOS FUTUROS

- Un entorno virtual de aprendizaje debe considerar herramientas para la generación de contenidos a nivel Web ya que esto permite crear contenidos educativos directamente sobre el entorno virtual, sin necesidad de utilizar herramientas de escritorio adicionales.
- La creación de contenidos educativos debe seguir algunas recomendaciones pedagógicas de usabilidad, accesibilidad, adaptabilidad y tecnológicas, que permiten que un contenido educativo sea de alta calidad, de fácil uso para los estudiantes, accesible a usuarios con distintas capacidades físicas y mentales, flexible en cuanto al estilo de aprendizaje de los estudiantes y que tenga un despliegue eficiente a nivel de Web.
- Existe gran cantidad de entornos virtuales de aprendizaje libres que poseen aplicaciones para el manejo de los diferentes procesos implicados dentro del e-learning, como seguimiento del estudiante, evaluación, diseño de actividades de aprendizaje, interacción síncrona y asíncrona entre el profesor y los estudiantes y entre estudiantes, etc. Estos entornos virtuales como dotLRN, Moodle, ATutor, etc. son maduros en muchos aspectos, por lo cual resultan una excelente opción a adoptar por parte de las instituciones educativas que desean ofrecer servicios de e-learning y/o b-learning, en un tiempo relativamente corto y con un nivel de calidad elevado y bajo costo. Otro aspecto que impulsa la adopción de este tipo de sistemas en una institución educativa es su experiencia con relación al e-learning, debido a que existen muchas instituciones a nivel mundial que han recorrido un largo camino en el mundo de la formación virtual, dando como resultado el desarrollo de algunos entornos virtuales de aprendizaje libres que plasman la gran evolución de dichas instituciones en el área del e-learning y que pueden impulsar en gran medida a una institución con baja experiencia en este campo.
- El planteamiento de lineamientos fundamentales para la creación de contenidos debe estar ligado al uso de especificaciones tendientes a estándares como es el caso de IMS-SCORM y LOM, ya que esto permite que los contenidos educativos se aproximen a

objetos de aprendizaje que puedan ser compartidos, reutilizados, almacenados en un repositorio de objetos de aprendizaje público, descargados de dichos repositorios, etc. Este tipo de procedimientos permiten el diseño y la creación de actividades de aprendizaje con el uso de contenidos educativos basados en objetos de aprendizaje reutilizables.

- La Universidad del Cauca no tiene mucha experiencia en el campo del e-learning en comparación con otras instituciones a nivel del país y con mayor razón a nivel latinoamericano y mundial. Por lo tanto proyectos como E-LANE y Unicauca Virtual permiten que esta institución educativa crezca y se fortalezca en el campo del e-learning que es un tema consolidado en otras instituciones y tiene gran proyección hacia futuro. Por esta razón, al interior de la Universidad del Cauca, es importante que se fomente el desarrollo de este tipo de proyectos.
- El sistema RACE es una buena iniciativa para la creación y gestión de contenidos educativos usables, adaptables y en menor medida accesibles y que tengan en cuenta la importancia de algunos aspectos tecnológicos como la velocidad de conexión, además de algunas recomendaciones pedagógicas esenciales. A pesar de todo, la iniciativa del sistema RACE no cubre en suficiente medida dichos aspectos, únicamente constituye una base. Por eso se recomienda que la experiencia adquirida en el análisis, diseño e implementación de RACE sea utilizada al momento de desarrollar contenidos educativos o herramientas que ofrezcan soporte a los mismos. No está demás tener en cuenta posibles extensiones del sistema RACE tendientes a una mayor funcionalidad en cuanto a los aspectos mencionados anteriormente.
- Un aspecto fundamental que no se alcanzó a cubrir a nivel de implementación fue la accesibilidad. Dado que se trabajó con contenidos con base en XML, esto brinda la posibilidad de procesar la información de dichos contenidos con mayor facilidad permitiendo pensar en una accesibilidad más universal. Una persona limitada visualmente podría acceder a contenidos en dotLRN con las indicaciones sonoras adecuadas como sintetizadores de voz, sonidos que indican la posición del cursor, enlaces sonoros, el diseño adecuado y las facilidades necesarias a nivel de hardware. Un discapacitado físico podría incluso acceder a contenidos adecuados a su “perfil”, en

dotLRN. En la facultad se han realizado trabajos a nivel de laboratorio y trabajos de pregrado relacionados con accesibilidad, lo cual también constituye un punto de partida hacia la construcción de un LMS accesible. Accesibilidad total implicaría brindar soporte también para dispositivos móviles, llegando a obtener una plataforma más accesible de lo que es actualmente, ampliando así el rango de perfiles a los que se puede llegar. Se propone como trabajo futuro que tomando como base el diseño e implementación de RACE se brinde soporte de mayor accesibilidad a un LMS como dotLRN, explorando al mismo tiempo los otros dos pilares de la interacción humano-computador (HCI) que son Usabilidad y Adaptabilidad.

- En un futuro el sistema RACE puede enfocarse a ser utilizado no solamente por el profesor sino también por el estudiante. El estudiante podría utilizar el editor para el desarrollo de sus tareas y trabajos y posteriormente colocarlos a disposición del profesor para su respectiva evaluación. Las tendencias educativas actuales buscan involucrar más al estudiante en el proceso de valoración de la calidad de los contenidos, por lo tanto el estudiante podría utilizar el editor para incluir sus propias observaciones y recomendaciones sobre el contenido desarrollado por el profesor, lo que podría conllevar a un mejoramiento de la calidad del contenido y un aumento en el nivel de aceptación del mismo.

BIBLIOGRAFÍA

- [1] Bello Díaz, Rafael Emilio. "Educación Virtual: Aulas sin paredes". 2005. Artículo de Internet. <http://www.educar.org/articulos/educacionvirtual.asp>
- [2] Amilcar. "E-learning". Octubre de 2003. Archivo PDF. http://www.tema-e.com.ar/E-learning/wp1_ed2.pdf
- [3] Pérez, Diego. "Situación actual del E-learning". Agosto del 2005. Artículo de Internet. http://www.webtaller.com/maletin/articulos/situacion_actual_del_elearning.php
- [4] Página Web del proyecto E-LANE. <http://e-lane.org>
- [5] Grupo de Ingeniería Telemática, Unicauca. <http://git.ucauca.edu.co/e-lane/>
- [6] Plataforma de teleformación dotLRN en Unicauca. <http://eva.unicauca.edu.co>
- [7] Zapata, Miguel. "Sistemas de Gestión del Aprendizaje – Plataformas de teleformación" 2003. Archivo PDF. <http://www.um.es/ead/red/9/SGA.pdf>
- [8] Página Web de la plataforma de teleformación Moodle. <http://moodle.org/mod/resource/view.php?id=684>
- [9] Página Web de la plataforma de teleformación ATutor. <http://atutor.ca/index.php>
- [10] Página Web de la plataforma de teleformación Dokeos. <http://www.dokeos.com/explain.php>
- [11] Página Web de la plataforma de teleformación Claroline. <http://www.claroline.net/>
- [12] Página Web del Framework de aplicaciones Web OpenACS. <http://www.openacs.org>
- [13] García Pozo, Pedro J. "Implantación de dotLRN en la Universidad de Valencia". Proyecto: Aula Virtual. Proyecto Fin de Carrera. Julio 2004. Documento PDF. <http://www.uv.es/ticape/docs/pedroj/proyecto.pdf>
- [14] Página Web de la plataforma de teleformación dotLRN. <http://www.dotlrn.org>
- [15] Calvo Rafael A., Ghiglione Ernie. "The dotLRN Management System Architecture". Universidad de Sydney - Australia. 2003. Archivo PDF. <http://www.weg.ee.usyd.edu.au/people/rafa/openacs/dotlrnArch.pdf>
- [16] TCL API, from AOLServer Wiki. <http://panoptic.com/wiki/aolserver/Tcl%20API>
- [17] Osmosis Latina. "Ventajas de AOLServer". 2005.

<http://www.osmosislatina.com/aolserver/ventajas.htm>

[18] Universidad de Oviedo. “Capítulo 1 TCL/TK”. 2003. Tutorial de TCL/TK. <http://www.etsimo.uniovi.es/tcl/tutorial/cap1.html>

[19] Lockhart Thomas. Tutorial de PostgreSQL. Octubre de 2001. <http://palomo.usach.cl/Docs/postgres/Postgres-Tutorial.pdf>

[20] González Arechavaleta Marta. “Como desarrollar contenidos para la formación online basados en objetos de aprendizaje”. Comunes, SA. 2004. Archivo PDF. http://spdece.uah.es/papers/GonzalezArechabaleta_Final.pdf

[21] Prendes Espinosa, Maria Paz. “Diseño de cursos y materiales para teleenseñanza”. Simposio Iberoamericano de Virtualización del aprendizaje y la Enseñanza, Costa Rica 2003. Archivo PDF. <http://tecnologiaedu.us.es/bibliovir/pdf/paz5.pdf>

[22] Gupta Arpita, P.S Grover. “Proposed Evaluation Framework for Adaptive Hypermedia Systems”. Proceedings of the Third Workshop on Empirical Evaluation of Adaptive Systems. Eindhoven University of Technology, The Netherlands, Agosto 23-26 de 2004.

[23] Bilgic Mustafa, Money Raymond J. “Explaining Recommendations: Satisfaction vs. Promotion”. Proceedings of Beyond Personalization 2005: A Workshop on the Next Stage of Recommender Systems Research at the 2005 International Conference on Intelligent User Interfaces, San Diego, CA, January 2005.

[24] P.P da Cruz, Rui Alexandre. García Peñalvo Francisco J. Romero Luis Alonso. “Perfiles de Usuario: en la senda de la personalización”. Enero 2003. Informe Técnico. <http://tejo.usal.es/inftec/2003/DPTOIA-IT-2003-001.pdf>

[25] De la Fuente, Valentín Luis. ¿Por qué docBook para generar contenidos?, Marzo de 2005, Documento PDF.

[26] <http://docbook.sourceforge.net/>

[27] Pérez Natalia. Tutorial de Edición de Contenidos con docbook. 2004.

[28] CESGA, Centro de Supercomputación de Galicia. “Estándares en e-learning”. <http://www.cesga.es/ca/defaultC.html?Teleensino/pt/pt-estandares.html&2>

[29] Borja Manero Iglesias. “Estudio de la propuesta IMS de estandarización de enseñanza asistida por computadora”. Informe técnico, Departamento de Sistemas Informáticos y Programación. Universidad Complutense de Madrid. Septiembre 2003. <http://www.fdi.ucm.es/profesor/borja/Informe%20tecnico.pdf>

[30] Álvarez Álvarez, José Valentín. “Uso de Estándares e-learning en espacios educativos” Universidad Metropolitana, Caracas – Venezuela. http://www.cica.es/aliens/revfuentes/campo_02.htm

[31] Foix Cristian, Zavando Sonia. Corporación de investigación tecnológica de Chile "INTEC", Centro de Tecnologías de Información. "Estándares e-learning, Estado del Arte". 10-07-2002.

<http://empresas.sence.cl/documentos/elearning/INTEC%20-%20Estandares%20elearning.pdf>

[32] Aguirre Sandra, Quemada Juan, Salvachua Joaquín. "Mediadores e Interoperabilidad en e-learning". Universidad Politécnica de Madrid, Departamento de Ingeniería de Sistemas Telemáticos. Octubre de 2004.

<http://jungla.dit.upm.es/~saguirre/publications/virtualEduca2004.pdf>

[33] Hernández Eduardo. Estándares y Especificaciones de e-learning.

http://www.ecampus.cl/Textos/tecnologia/eduardo_hernandez/eduardo.htm

[34] Página oficial de la organización ADL, desarrolladora de SCORM.

<http://www.adlnet.org/aboutadl/index.cfm>

[35] Hernández Luis Eduardo. Ponencia: Centro de Educación a Distancia de la Universidad Católica del Norte. "Estado Actual y Futuro de las Plataformas e-learning", Estándares y Especificaciones IMS. Agosto de 2002.

<http://fad.uta.cl/dfad/docum/cedm/1-cl-Luis%20Hern%E1ndez%20Sc%E4fer-Gim%20lvy.pdf>

[36] http://www.adlnet.org/adl_structure/

[37] Modelo de referencia de objetos de contenido compatible

<http://www.adlnet.org/scorm/index.cfm>

[38] Dieguez Jorge, Calvet Albert. E-learning Workshops. Academia de e-learning. "SCORM y YO: Curso de introducción a SCORM". 2004.

<http://www.academiaelearning.com/contenido/scorm/scormyyo/sco/curso.htm>

[39] SCORM Content Aggregation Model. Version 1.3.1. Julio 22, 2004.

http://www.adlnet.org/SCORM_CAM.pdf

[40] SCORM Run-Time Environment. Version 1.3.1. Julio 22, 2004.

http://www.adlnet.org/SCORM_RunTimeEnv.pdf

[41] <http://www.oasis-open.org/docbook/xmlschema/>

[42] <http://es.wikipedia.org/wiki/Wysiwyg>

[43] <http://es.wikipedia.org/wiki/WYSIWYM>

[44] V. Fresno-Fernández, S. Montalvo-Herranz, J. Pérez-Iglesias, J.A. Velazquez-Iturbide. "eXitor: a tool for the assisted edition of XML documents". Seventh International Conference on Electronic Publishing. Guimaraes, Portugal University of Minho. 25-28 June 2003. pp 308-315. ISBN 972-98921-2-1.

[45] Turbo XML. <http://www.tibco.com>

- [46] XMetal. <http://www.softquad.com/products/xmetal>
- [47] XMLSpy. <http://www.xmlspy.com>
- [48] Amaya. <http://www.w3c.org/amaya>
- [49] HTML-KIT. <http://www.chami.com/html-kit>
- [50] XMLEditPro. <http://www.daveswebsite.com>
- [51] XMLOperator. <http://www.xmloperator.net>
- [52] Course Composer. <http://www.myrnham.co.uk>
- [53] Morphon XML-Editor. <http://www.morphon.com>
- [54] XMLMind. <http://www.xmlmind.com/xmleditor>
- [55] <http://quanta.kdwebdev.org/>
- [56] <http://www.butterflyxml.org/>
- [57] <http://sourceforge.net/projects/tclxml>
- [58] <http://www.vervet.com>
- [59] <http://www.w3.org/XML/Schema>
- [60] <http://media.wu-wien.ac.at/download/xowiki-doc/index.html>
- [61] <http://www.reload.ac.uk/editor.html>
- [62] <http://weload.lernnetz.de/>
- [63] Gallego Rodríguez Alejandrino, Martínez Caro Eva. Estilos de aprendizaje y e-learning. Hacia un mayor rendimiento. <http://www.um.es/ead/red/7/estilos.pdf>
- [64] <http://www.monografias.com/trabajos12/loestils/loestils.shtml>
- [65] <http://www.fce.unju.edu.ar/download/si/APRENDIZAJE.pdf>
- [66] <http://www.iese.edu/mail/davidkolb.html>
- [67] <http://validator.w3.org/>
- [68] http://es.wikipedia.org/wiki/Tabla_comparativa_de_navegadores_web
- [69] <http://home.cfl.rr.com/ea/Bandwidth.htm>

[70] http://reviews-zdnet.com.com/Bandwidth_meter/7004-7254_16-0.html

[71] Jongerius, Jerry. A comparison of Internet connection speed testers, Jerry. March 4, 2005 – v16. <http://www.myspeed.com/support/benchmarks.html>

[72] <http://www.auditmypc.com/internet-speed-test.asp>

[73] Villa Luis. Empleo del color. 14-01-2003. http://www.alzado.org/articulo.php?id_art=12

[74] Villa, Luis. Artículo “Empleo del color”. Enero de 2003. http://www.alzado.org/articulo.php?id_art=12

[75] Manchon, Eduardo. Artículo “Evita los colores de fondo”. Julio de 2004. http://www.alzado.org/articulo.php?id_art=343

[76] De La Fuente Valentín, Luis. Universidad Carlos III de Madrid. Tutorial para creación de documentos en Docbook y empaquetamiento según content packaging de SCORM utilizando la herramienta Reload. 2005. Documento no público proyecto E-LANE.

[77] Archibald Tang, Beth. Artículo “Accessibility and Usability: What's In It For Me”. Junio de 2000. <http://www.webdevelopersjournal.com/articles/accessibility.html>

[78] ETSI DEG 202 417 v0.0.30 (2006-03-10): "Human Factors; User education guidelines for mobile terminals and e-services".

[79] <http://www.htmlhelp.com/es/design/accessibility/why.html>

[80] <http://www.htmlhelp.com/tools/validator/>

[81] <http://www.delorie.com/web/purify.html>

[82] <http://www.htmlhelp.com/es/design/accessibility/tips.html>

[83] <http://www.delorie.com/web/wpbcv.html>

[84] Moreno, Luciano. “Que es la usabilidad? Como se mide la usabilidad de un sitio Web?”. http://www.htmlweb.net/usabilidad/usabilidad_1/usabilidad_1_6.html

[85] Collazos César A, Baloian Nelson, Galdames Patricio, Guerrero Luis A. Artículo, “A Model for a Collaborative Recommender System for Multimedia Learning”. Universidad de Chile - Universidad del Cauca. 2004.

[86] Fernández Panadero, M^a Carmen. Objetos de Aprendizaje, Primeros Pasos – Diseño de Actividades de Aprendizaje. Videoconferencias para el equipo E-LANE de la Universidad del Cauca desde la Universidad Carlos III de Madrid. 13 y 20 de Junio de 2005.

[87] Fernández Panadero, M^a Carmen. EPM Un modelo para la Caracterización y Diagnóstico de procesos Educativos. Tesis Doctoral. Departamento de Ingeniería Telemática, Escuela Politécnica Superior, Universidad Carlos III de Madrid. 2004.

[88] <http://www.adlnet.org/scorm/index.cfm>

[89] <http://www.openacs.org/xowiki/>

[90] http://www.alzado.org/articulo.php?id_art=54

[91] Palta Velasco, Eleonora. "Guía metodológica para el desarrollo de aplicaciones multimediales de apoyo a procesos de enseñanza-aprendizaje". Fundación Universitaria de Popayán, Colombia. 2000.

[92] CVS de OpenACS: <http://cvs.openacs.org/cvs/openacs-4/packages/#dirlist>

Lista de nuevas características y animaciones de las aplicaciones de dotLRN:

<http://www.dotlrn.org/demos-dotlrn/>

Sistema de Gestión de actividades de aprendizaje LAMS:

<http://www.lamsinternational.com/>

Beehive, aplicación software para aprendizaje colaborativo síncrono:

<http://www.weg.ee.usyd.edu.au/people/aimant/paper2.pdf>

[93] Serrano Castaño Carlos Enrique, Solarte Sarasty Mario Fernando, Ramírez González Gustavo Adolfo. Referencia Metodológica Integral para Desarrollo de Sistemas Telemáticos. Grupo en Ingeniería Telemática, Facultad de Ingeniería Electrónica y Telecomunicaciones, Universidad del Cauca. Popayán, Noviembre de 2002.

ftp://jano.ucauca.edu.co/proyectos/Ambientes_de_Desarrollo/Trabajos_en_Infraestructura_Metodologica/R3_RMI-Desarrollo-ST-V1.zip

[94] Conallen Jim. Web Application Extension for UML.

<http://www.rationalrose.com/stereotypes/waesetup.htm>

[95] Herramienta para modelado software a nivel Web.

<http://www.gskinner.com/gmodeler/app/run.html>

[96] Rendón, Alvaro. Lenguaje Unificado para Construcción de modelos. Curso publicado a través del entorno virtual de aprendizaje EVA.

[97] Booch Grady, Rumbaugh James, Jacobson Ivar. El Lenguaje Unificado de Modelado. Addison Wesley. 1999.

[98] Claros Ivan Dario. Usabilidad en sitios Web. Trabajo de Grado. Facultad de Ingeniería Electrónica y Telecomunicaciones. Universidad del Cauca. 2006.

[99] Faundez Fabiola, Labbé Carmen Gloria. Guía de Buenas prácticas para iniciativas de capacitación en modalidad e-learning. Junio de 2004.